UNIVERSIDAD COMPLUTENSE DE MADRID

FACULTAD DE INFORMÁTICA

Departamento de Ingeniería del Software e Inteligencia Artificial

TESIS DOCTORAL

Computer vision techniques for greenness identification and obstacle detection in maize fields

Técnicas de visión por computador para la detección del verdor y la detección de obstáculos en campos de maíz

MEMORIA PARA OPTAR AL GRADO DE DOCTOR

PRESENTADA POR

Yerania Campos Silvestre

Directores

Gonzalo Pajares Martinsanz
Humberto Sossa Azuela

Madrid, 2018

Computer vision techniques for greenness identification and obstacle detection in maize fields

Técnicas de visión por computador para la identificación del verdor y la detección de obstáculos en campos de maíz

Tesis Doctoral

Yerania Campos Silvestre

Departamento de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática
Universidad Complutense de Madrid

Computer vision techniques for greenness identification and obstacle detection in maize fields

by

Yerania Campos Silvestre

Dirigida por los Doctores
Supervised by

Gonzalo Pajares Martinsanz and Humberto Sossa Azuela

Memoria que presenta para optar al título de
Doctor en Informática
A thesis submitted in partial fulfillment for the degree of
PhD in Computer Science

in the

Departamento de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática
Universidad Complutense de Madrid

Madrid, 2017

Acknowledgements

This work was carried out during the years 2013-2016 at the Facultad de Informática, at the Departamento de Ingeniería del Software e Inteligencia Artificial, Universidad Complutense de Madrid, Spain. It has been a period of intense learning, not only in the scientific area, but also on a personal level. I would like to reflect on the people and sponsoring institutions who have supported and helped me so much throughout this period.

Foremost, I owe my deepest gratitude to my advisors Professor **Gonzalo Pajares Martinsanz** and Professor **Juan Humberto Sossa Azuela**, without whose continuous encouragement and support, this study would hardly have been completed. Thanks for their patience, motivation, and immense knowledge. Their guidance and supervision has been essential during this work. Without their precious support it would not be possible to conduct this research.

I am deeply grateful to Univ.-Pro I, Dr. **Joachim Denzler**, Head of the Institute of Informatics, Friedrich-Schiller-University Jena, for making it possible to carry out a research stay at his Computer Vision Group. Particularly, I am grateful to Dr. **Erik Rodner** for his valuable guidance. He, definitely provided me with the tools that I needed to continue my research during my stay in Jena University.

My sincere thanks also goes to Univ.-Pro I, Dr. **Oliver Bimber**, Head of the Institute of Computer Graphics at Johannes Kepler University Linz (JKU), who offered me the opportunity to join their team, and who gave access to the laboratory and research facilities. His guidance and suggestions were fundamental for the improvement of my research skills. Thanks for his valuable friendship and support.

I am also grateful to Dr. **Alejandro Castillo Mendez** from the Institute of Nuclear Research (ININ), Mexico. I am extremely thankful and indebted to him for sharing his expertise and sincere and valuable guidance and encouragement extended to me.

I would also like to thank the experts and members of my committee who were involved in the validation survey for this thesis. Without their participation, encouragement, insightful comments, and hard questions, my work could not have been successfully conducted.

I wish to thank all my friends and colleagues at UCM, Jena University, JKU, and ININ, researcher groups for their feedback, cooperation, favours and friendship. They all have been great in helping me out and I was able to have lots of fun with them.

Thanks to The National Council of Science and Technology, **CONACyT**, Mexico, in conjunction with The National Council of Science and Technology, **COMECyT**, Estado de Mexico, for the doctoral grant number 210282 to undertake doctoral studies and for the economic support, under project 65 (Frontiers of Science) to present my work at ACIVS 2016.

Additionally, I would like to express my gratitude to COMECyT, for the support received to perform my two research stays in the ININ, Mexico, during my PhD studies.

Thanks to the European Union's Seventh Framework Programme (FP7/2007-2013) under Grant Agreement No. 245986, the research leading to these results has been part of this project.

Finally, I also place on record, my sense of gratitude to one and all, who directly or indirectly, have lent their hand in this venture.

Thank you very much, everyone!

Yerania Campos Silvestre

Madrid, Spain, 2017

To my family; Ramón, Caty, Dalila, Ariel and Bastian.
I love you!

Contents

List of Figures

List of Tables

Abbreviations

Acronym	
AAV	Autonomous Agricultural Vehicles
PA	Precision Agriculture
RHEA	Robot fleets for highly effective and forestry management
AI	Agricultural Image
BoW	Bag of Words
CVI	Colour Vegetation Indices
SVM	Support Vector Machines
KNN	K-Nearest Neighbours
LIDAR	Light Detection And Ranging
ROI	Regions Of Interest
KM	K-Means
SOM	Self Organization Maps
FCM	Fuzzy C-Means
OS	Over-Segmentation
SIFT	Scale-Invariant Feature Transform
SURF	Speed-Up Robust Features
CNN	Convolutional Neural Networks
RFT	Random Forest Tree
SDC	Static and Dynamic Combined approach
OF	Optical Flow
TD	Temporal Difference
MoG	Mix of Gaussians

UNIVERSIDAD COMPLUTENSE DE MADRID

Abstract

Facultad de Informática

Departamento de Ingeniería del Software e Inteligencia Artificial

PhD in Computer Science

by Yerania Campos Silvestre

There is an increasing demand in the use of Computer Vision techniques in Precision Agriculture (PA) based on images captured with cameras on-board autonomous vehicles. Two techniques have been developed in this research. The first for greenness identification and the second for obstacle detection in maize fields, including people and animals, for tractors in the RHEA (robot fleets for highly effective and forestry management) project, equipped with monocular cameras on-board the tractors.

For vegetation identification in agricultural images the combination of colour vegetation indices (CVIs) with thresholding techniques is the usual strategy where the remaining elements on the image are also extracted. The main goal of this research line is the development of an alternative strategy for vegetation detection. To achieve our goal, we propose a methodology based on two well-known techniques in computer vision: Bag of Words representation (BoW) and Support Vector Machines (SVM). Then, each image is partitioned into several Regions Of Interest (ROIs). Afterwards, a feature descriptor is obtained for each ROI, then the descriptor is evaluated with a classifier model (previously trained to discriminate between vegetation and background) to determine whether or not the ROI is vegetation.

The descriptor used to represent ROIs is obtained with the BoW strategy; this is a sparse vector of occurrence counts of a visual vocabulary of local image features. The visual vocabulary is usually obtained by quantifying the image features into visual words. Feature selection plays an important role in the classifier performance function.

The research includes a comparative analysis of different features to find the best to characterize vegetation; this is another goal of this research. Features obtained by mixing different CVIs reported in literature were extracted for green plants characterization in maize fields in this research, outperforming SIFT (Scale-invariant feature transform) and SURF (Speed-up robust features) descriptors, as shown by the experimental results. Their performance (under different illumination conditions and growth plant stages) is similar to those methods reported in current literature. The classification accuracy achieved is over 95%; however, the segmentation method needs additional improvements. In this context, an accuracy of 86% was obtained; this result is because although the classifier achieves good performance, the segmentation algorithm depends on the method used to obtain the ROIs in the image. It is important to mention that 86% is comparable with the results obtained with other methodologies applied to our image dataset.

In this work, obstacle detection is carried out from the analysis of image sequences; this is the main aim of this research line. A new strategy for automatic image analysis is proposed to detect static/dynamic obstacles in agricultural environments via spatial-temporal analysis. At a first stage obstacles are detected by using spatial information based on spectral colour analysis and texture data. At a second stage, temporal information is used to detect moving objects/obstacles at the scene. The proposed method does not require any training process, therefore making an important contribution. Another contribution consists of the spatial analysis to obtain an initial segmentation of objects of interest; afterwards, temporal information is used for discriminating between moving and static objects. Usually, in agricultural image analysis, classical approaches make use of either spatial or temporal information, but not both simultaneously, making another important contribution. The proposed approach performs favourably when tested on different outdoor scenarios in agricultural environments achieving an obstacle detection rate of 93.88%. For motion detection, positive results were obtained to detect static and moving obstacles with the tractor stopped achieving an average accuracy of 95%, however, this value decreases significantly when the tractor is moving by achieving a poor performance rate of 66%. Still, it is difficult to identify if the motion comes from the tractor in motion or from dynamic objects, requiring an additional effort in order to improve this result.

To conclude, methods developed were widely tested under different scenarios which are really complex, mainly due to the high variability in the illumination conditions. Also,

the analysis included a comparative study of performance between the proposed methods and existing methodologies. The experimental results demonstrated that the methods here proposed are good alternatives to be considered in the autonomous agricultural vehicles design.

Keywords: Precision agriculture, dynamic/static obstacle detection, vegetation detection, autonomous vehicles, texture analysis, temporal analysis, greenness segmentation, temporal analysis, interest regions, feature selection, local descriptors, colour vegetation indices.

UNIVERSIDAD COMPLUTENSE DE MADRID

RESUMEN

Facultad de Informática
Departamento de Ingeniería del Software e Inteligencia Artificial

Doctor en Informática

por Yerania Campos Silvestre

Cada vez existe mayor demanda en el uso de técnicas de Visión por Computador en Agricultura de Precisión mediante el procesamiento de imágenes captadas por cámaras instaladas en vehículos autónomos. En este trabajo de investigación se han desarrollado dos tipos de técnicas. Una para la identificación de plantas verdes y otra para la detección de obstáculos en campos de maíz, incluyendo personas y animales, para tractores del proyecto RHEA. El objetivo final de los vehículos autónomos fue la identificación y eliminación de malas hierbas en los campos de maíz.

En imágenes agrícolas la vegetación se detecta generalmente mediante índices de vegetación y métodos de umbralización. Los índices se calculan a partir de las propiedades espectrales en las imágenes de color. En esta tesis se propone un nuevo método con tal fin, lo que constituye un objetivo primordial de la investigación. La propuesta se basa en una estrategia conocida como "bolsa de palabras" conjuntamente con un modelo se aprendizaje supervisado. Ambas técnicas son ampliamente utilizadas en reconocimiento y clasificación de imágenes. La imagen se divide inicialmente en regiones homogéneas o de interés (RIs). Dada una colección de RIs, obtenida de un conjunto de imágenes agrícolas, se calculan sus características locales que se agrupan por su similitud. Cada grupo representa una "palabra visual", y el conjunto de palabras visuales encontradas forman un "diccionario visual". Cada RI se representa por un conjunto de palabras visuales las cuales se cuantifican de acuerdo a su ocurrencia dentro de la región obteniendo así un vector-código o "codebook", que es descriptor de la RI. Finalmente, se usan las Máquinas de Vectores Soporte para evaluar los vectores-código y así, discriminar entre RIs que son vegetación del resto.

En la propuesta se incluyeron descriptores que se obtienen a partir de la combinación de varios índices de vegetación, los cuales obtuvieron un mejor rendimiento que los

descriptores SIFT y SURF ampliamente utilizados. Se realiza además un análisis comparativo sobre descriptores.

Los resultados obtenidos muestran que el método es suficiente para identificar las plantas bajo diferentes condiciones de iluminación y crecimiento de las mismas. Se alcanzó una tasa de clasificación de 95% para discriminar plantas del resto de los elementos en una imagen. El método en conjunto obtuvo una tasa de acierto del 86% para la segmentación de la vegetación, un valor semejante a los obtenidos con otras técnicas reportadas en la literatura actual.

Con respecto a la detección de obstáculos, se propone un método basado exclusivamente en la información contenida en imágenes agrícolas consecutivas, obtenidas mediante un sistema de visión monocular. El método ha sido diseñado para detectar y discriminar entre obstáculos estáticos y objetos en movimiento, siendo éste el objetivo fundamental del trabajo de investigación desarrollado en esta línea. En ambos casos, se trata de un método de ayuda a la navegación para evitar colisiones. Las plantas verdes y el suelo constituyen el punto de partida de la propuesta. En una primera fase, se obtienen propiedades de textura y espectrales (color) en el instante de tiempo t, las cuales se procesan con métodos de umbralización para identificar los elementos no habituales en el campo de cultivo, es decir, aquellos que no son plantas ni suelo. Posteriormente, para discriminar entre elementos estáticos de los no estáticos se hace uso de la derivada temporal, sin necesidad de aprendizaje, suponiendo un avance respecto de los métodos existentes a la vez que un objetivo específico.

La validación de la técnica se hizo en dos partes, primero se evaluó la segmentación de los objetos y posteriormente se probó la eficacia del método para discriminar entre obstáculos estáticos y dinámicos. En la primer fase, el algoritmo de segmentación resultó eficiente alcanzando una tasa de detección de 93.88%. Para la detección de movimiento, los resultados se presentan favorables para discriminar entre obstáculos estáticos y en movimiento cuando el tractor está parado, con un valor de detección promedio de 95%. Sin embargo, el método presenta fallos cuando el tractor y los objetos están en movimiento obteniendo una tasa de discriminación de 66%. En este sentido, aún es difícil determinar si el movimiento proviene del vehículo o de los objetos, se requiere un esfuerzo adicional para mejorar los resultados bajo este escenario.

En conclusión, se han diseñado métodos eficientes para la detección de verdes y la detección de obstáculos en el contexto de las imágenes obtenidas durante el desarrollo del proyecto RHEA, que han demostrado su eficiencia por comparación con otras estrategias probadas en el contexto agrícola. En ambos casos los métodos aquí expuestos son una opción que pueden ser considerados durante el diseño de vehículos agrícolas autónomos.

Palabras clave: Agricultura de precisión, detección de obstáculos estáticos y dinámicos, detección de vegetación, vehículos autónomos, análisis de textura, análisis temporal, segmentación de verdes y suelo, regiones de interés, selección de características, descriptores locales, índices de vegetación.

Chapter 1

Introduction

Agriculture is one of the oldest human activities. Over the years, farmers developed methods of preparing the soil and protecting the plants not only from environmental problems like storm, heavy rain, pests, as well as other problems like delay in expert advice of disease, nutrients deficiency problem, limited resources like water and electricity. As a result, the use of fertilizers and cost increased significantly. Nowadays, the use of agricultural machines to reduce the labour required is common, moreover, several researches have been attracted to this field with the aim of increasing quality and quantity of agricultural crops. In particular, computer science researchers are attracted towards agriculture fields to increase accuracy automation in robotic technology. For reducing manual labour work, computer technologies play an important role in the automation of robotics where advances in electronics, artificial intelligence, machine vision and other methods and technologies have been integrated in the design and development of Autonomous Agricultural Vehicles (AAV) to perform a wide range of activities in agricultural tasks (planting seeds, fertilizing, applying pesticides, pruning, thinning, tilling, and harvesting, as well as mowing, spraying, and weed removal) increasing the efficiency and reducing the cost of agricultural production and derived products [11]. In evolution towards sustainable agriculture systems it was clear that important contributions were possible by using emerging technologies, where imaging-based approaches are called to play an important role. This is the kernel of the research described in this document.

1.1 Precision agriculture

An important concern in agriculture is productivity where agricultural tasks have to be carried out with accuracy, maximum performance, and minimal resources. This

situation means that the integration of all involved systems in AAV, including the vehicle and the implement for acting, must be conveniently structured under a specific architecture with an effective and reliable design to meet all these requirements. This task is commonly known and accepted as Precision Agriculture (PA), which can be defined as the application of correct amount of inputs (water, fertilizer, pesticides, etc.) at the correct time. Because of this, PA has become a cornerstone in agricultural applications where the use of robots, equipped with vision-based sensors, have been seen in continuous growth to apply the right treatment at the right place and at the right time [12].

The following are the main benefits of PA: a) contributes to the wider goal concerning sustainability of agricultural production; b) minimizes the use of machinery, input costs, use of water, application of fertilizers (avoids overlapping during application of fertilizers and pesticides), and uncertainty for making decisions required to manage variability on farms; and c) maximizes profitability, productivity, sustainability, crop quality, environmental protection, on-farm quality of life, food safety, and rural economic development. Summarizing, the application of the right amount of inputs in the right place and at the right time benefits crops, soil and groundwater, and thus the entire crop cycle increasing crop quality and productivity.

In later years, advanced techniques have been incorporated to enhance farm output and also to enrich the farm inputs in profitable and an environmentally sensible manner [13]. Although it is true that technology in PA has been proven efficient, there are still several important issues that must be solved or improved [14], requiring methods to optimize the product quality and quantity while minimizing cost, human intervention, environment damage (avoiding the undesirable effects of excess chemical loading to the environment), and the variation caused under unpredictable environmental conditions.

1.2 Autonomous vehicles

The design of autonomous vehicles for agricultural applications is becoming more frequent; they must be equipped with perception systems (for acquiring information from the environment), decision-making systems (for interpreting and analysing such information), and actuation systems (responsible for performing the agricultural operations). In turn, these systems consist of different sensors, actuators, and methods, procedures and algorithms running in computers that work synchronously in a specific architecture for the intended purpose. An example of this architecture can be seen in Fig. 1.1, where an autonomous tractor as the main system is equipped with perception (sensors), decision and actuation subsystems prepared to work on real wide-row crops;

it was designed to apply specific treatments in crop maize fields. The tractor is part of the fleet in the RHEA project [1].

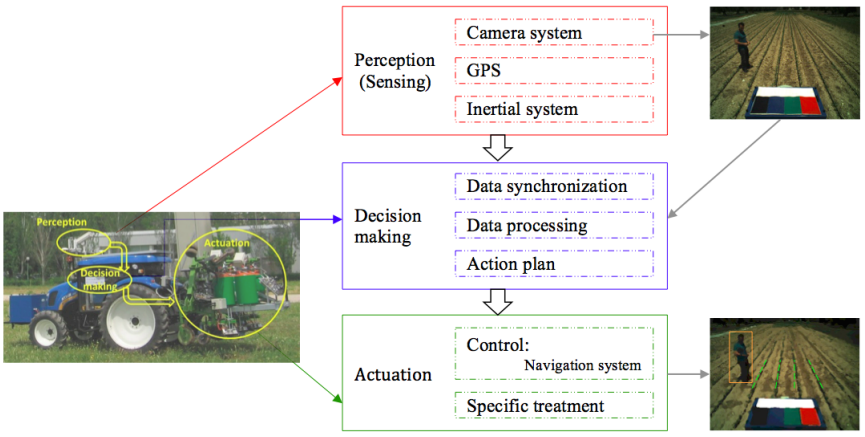


Figure 1.1: Design of an autonomous tractor, it is part of the fleet in the "Robot Fleets for Highly Effective Agriculture and Forestry Management (RHEA)" project [1].

Several designs of AAV have been proposed since 1960; all with of them with the aim to perform the different agricultural tasks above mentioned reducing human effort and minimizing costs [15–19]. Even thought these technologies maximized overall productivity, they led some areas within fields to underperform. The development of this research work is based on the analysis of images acquired with the perception system, and more specifically with the machine vision system designed for the autonomous tractors belonging to the fleet on the RHEA project [1], the vision system is described in Chapter 2.

In the following sections, the main objectives of this work are proposed.

1.3 Objectives

Based on the above considerations, this work is focused on analysing the agricultural images (AI) obtained with the machine vision system described in Section 2.2 containing two main objectives: vegetation segmentation and obstacle detection. The first is useful for weed and crop row identification, the second is important for safety reasons to prevent collisions in the working area ensuring the process continuity. These problems can be seen as a segmentation process where the agricultural image must be split into multiple segments or categories. For example, Fig. 1.2 displays two original images and their respectively segmented versions where each colour represents a component: green plants, soil, vehicles, people, structures, sky, fence post, etc. From the point of view of image processing, achieving a segmentation similar to the images in Fig. 1.2 still represents a difficult problem mainly in images coming from outdoor environments. Then, to achieve

our two main objectives two methods were proposed making the main contribution of this work.

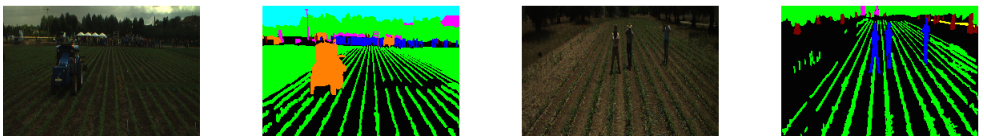


Figure 1.2: Components identification in agriculture images; greenness segmentation is useful for crop row identification and weed removal, meanwhile obstacles on the scene must be detected to avoid collisions.

Vegetation segmentation

An alternate methodology for vegetation segmentation against classical approaches [6, 20–27] in images coming from maize fields is provided; the process includes two main steps: a low-level segmentation and a class label assignment using Bag of Words representation in conjunction with a supervised learning framework. Several experiments were conducted to evaluate the method efficiency. The study included a texture analysis and a comparison against several strategies well known in current literature, detailed in Chapter 3. As above-mentioned, the machine vision system on the RHEA tractors is focused on crop row detection in a specific area in front of the tractor. In this work the rectangular area in front of the tractor with size 450×920 pixels, green dotted lines in Fig. 1.3, was considered for analysis.

Obstacle detection

A new strategy for automatic video analysis to detect static/dynamic obstacles in agricultural environments, via spatial-temporal analysis, is proposed to ensure the safety and care of people or animals in the field, but especially those on the tractor trajectory. In addition, not only static obstacles detection is of interest, also, we study obstacles which are moving in the crop area. Motion analysis is especially valuable for those elements which can suddenly appear in front of the vehicle. Detection consists in an initial segmentation of objects of interest; afterwards, temporal information is used for discriminating between moving and static objects. For this task, the area under analysis must be larger than the one used for vegetation analysis; it was expanded covering five crop rows 900×1530. Such dimensions were defined trying to cover the maximum area in the image to guarantee that close objects to the autonomous vehicle were detected, dotted yellow lines in Fig. 1.3.

As a result of the research carried out to achieve the aims above described, the following list of papers provide direct support to the research described here.

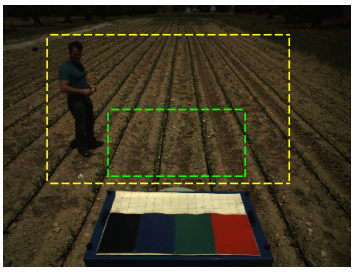


Figure 1.3: Colour images of size 1752×2336; vegetation detection is detected in the area enclosed in yellow lines, for obstacle detection, the area under analysis is the rectangular area limited by the yellow lines.

1. **Campos Yerania**, Rodner Erik, Denzler Joachim, Sossa Humberto and, Pajares, Gonzalo. "Vegetation Segmentation in Cornfield Images Using Bag of Words". *Advanced Concepts for Intelligent Vision Systems: 17th International Conference*, ACIVS 2016, Lecce, Italy, October 24-27, 2016, Proceedings (193-204). Springer International Publishing. ISBN: $978-3-319-48680-2$. DOI: $10.1007/978-3-319-48680-2\_18$. *Abstract*: We provide an alternative methodology for vegetation segmentation in cornfield images. The process includes two main steps which make the main contribution to this approach: a) a low-level segmentation and b) a class label assignment using Bag of Words (BoW) representation in conjunction with a supervised learning framework.

2. **Campos Yerania**, Sossa Humberto, and Pajares, Gonzalo. "Comparative analysis of texture descriptors in maize fields with plants, soil and object discrimination". *Precision Agriculture*, 2016(1-19). SNN:1573 – 1618, DOI:$10.1007/s11119-016-9483-4$. *Abstract*: The objective of this study was to design a tri-class Support Vector Machine classifier for identifying plants (crops and weeds), soil and objects in maize fields based on unsupervised learning. The analysis included a comparative strategy of different texture descriptors and local patterns to determine the best feature descriptor for characterizing the classes under study.

3. **Campos Yerania** and Sossa Humberto and Pajares Gonzalo. "Spatio-temporal Analysis for Obstacle Detection in Agricultural Videos". *Applied Soft Computing*. August 2016. Volumen 45. Pages (86-97). Elsevier Science Publishers B. V., Amsterdam, The Netherlands, The Netherlands. DOI = $10.1016/j.asoc.2016.03.016$. *Abstract*: We propose a new strategy for automatic video analysis to detect static/dynamic obstacles in agricultural environments via spatial-temporal analysis. At a first stage obstacles are detected by using spatial information based on spectral colour analysis and texture data. At a second stage temporal information is used to detect moving objects/obstacles at the scene, which is of particular interest in camouflaged elements within the environment.

4. **Campos Yerania** and Sossa Humberto and Pajares Gonzalo. "Detección de obstáculos en imágenes de campos de maíz". *Revista Internacional de Investigación e Innovación Tecnológica*. Julio – Agosto 2016. Volumen 4, No. 21. ISSN: 2007 – 9753. Latindex Folio: 23614. *Abstract*: Detection of unexpected obstacles in the path of an autonomous agricultural vehicle is our main aim. In this paper, we propose a strategy for automatic obstacle detection combining image processing techniques and supervised learning algorithms.

Additionally, the paper of Pajares et al. [19] provides a guide for machine vision systems selection in agricultural environments, where obstacle detection is considered as part of the whole system together with considerations about spectral band selection, sensors and optical systems and geometrical system arrangement.

Pajares Gonzalo and García Iván and **Campos Yerania** and Montalvo Martín and Guerrero José M. and Emmi Luis and Romeo Juan and Guijarro María and Gonzalez-de-Santos Pablo. "Machine-Vision Systems Selection for Agricultural Vehicles: A Guide." J. Imaging 2, no. 4: 34. 2016.

To conclude this chapter, the thesis organization is as follows: Chapter 2, according to the two main objectives, provides a revision about the state-of-the art involving machine vision systems in AAV focused on vegetation segmentation and obstacle detection. The proposed methodology for vegetation segmentation with comparative analysis is presented in Chapter 3. Chapter 4 describes the proposal and performances of obstacle detection. To close our work, the general conclusions are given in Chapter 5.

Chapter 2

Background

In this chapter a review of the state-of-the-art in agricultural autonomous vehicles is proposed. According to the two main objectives provided in the previous chapter, the review is focused specially on works where "vegetation detection" and "obstacle detection" in agricultural applications were the main research topics. Also, the vision system in the RHEA tractors is described due to it being the basis for the analysis of methodologies described in Chapters 3 and 4.

2.1 State-of-the-art

Autonomous navigation systems for mobile robots are making an increasing presence in agriculture applications. The first attempts to develop AAV were reported in the sixties, Li et al. [15] provided a research review from 1960s till 2009 on guidance systems and technologies in agricultural vehicles for the commercialization of the guidance system. The rapid growth in computing technologies has provided important progress in this field [28, 29] introducing new technologies to increase the effectiveness of the navigation systems. These proposals are summarized by Mousazadeh [17], Vibhute et. al [30] and Saxena [31].

2.1.1 Machine vision systems

Different image acquisition systems have been proposed in literature with the aim of analysing the images from different crops fields and different purposes: identification of weed species [32], mature wheat detection [33], weed elimination [34], crop row detection [35], pest identification [36], estimation of plants nitrogen content [37], and many other applications.

The vision system may consist of a single camera (monochrome or colour), multiple cameras (e.g., stereo vision), multispectral imaging or range sensing devices with image analysis algorithms developed to identify specific features in the crops, weeds, soil and perhaps other elements of interest, including obstacles under a broad range of adverse environmental conditions in outdoor scenarios [38]. For example; a) high-resolution images are acquired with cameras installed on a high-flying platform (satellites, airplanes, balloons and helicopters) or with unmanned aircraft system. These aerial images have been used to identify field variations, crop species classification and crop pest management [39]; b) thermal images where the invisible radiation pattern of an object transformed to visible images have been used for predicting water stress in crops, planning irrigation schedule, disease and pathogen detection in plants, predicting fruit yield, evaluating maturing fruits, bruise detection in fruits and vegetables, detection of foreign bodies in food material, and temperature distribution during cooking [40]; c) X-ray images have also been used to assess the tomato quality, wheat kernels and walnuts and to detect defects, toxic metals and nutrients [41, 42]; d) computed tomography images were applied to study the soil and its physical structures [43]. From the different image acquisitions McCarthy et al. [38] argued that adding a monocular RGB vision system with additional sensing techniques potentially reduces image analysis complexity while enhancing system robustness against adverse environmental conditions and variability.

From the analysis of AI captured with machine vision-based systems such as those described above, the following paragraphs provide a literature review in vegetation segmentation researches in agricultural environments.

2.1.2 Vegetation segmentation

Haug et al. [44] proposed a machine vision approach to discriminate crops and weeds of carrot images achieving an accuracy value of 93.8%. Images were acquired with a multi-spectral down–looking monocular camera in the visible red $\tilde{R}$ and near-infrared (NIR) spectrum. Soil is removed in [45] by applying the Normalized Difference Vegetation Indices (NDVI) [46], to each pixel in the input image. A threshold is selected in NDVI space using Otsu's method and all pixels with NDVI values smaller than the threshold are masked. Hlaing and Khaing [21] developed a method to discriminate among crops and weeds having a misclassification rate of 33.3%. The absolute values of green minus red and green minus blue are calculated at each pixel in the image. If both of these values are greater than a threshold value, the pixel is classified as plant, otherwise, the pixel is classified as background. Plant pixels are evaluated with a decision function provided by a classifier to discriminate between crops and weeds.

Tewari et al. [20] designed and developed a microcontroller for site-specific herbicide application with an automatic weed detection technique in crop field images. For weed detection, each pixel in the image is processed as follows. When green colour intensity is greater than red as well as blue colour intensities, the pixel is assumed to be green pixel, otherwise, the pixel is assumed to be background. The total number of green pixels is divided by the image size; this value represents the percentage of weeds in the image. They reported a weeding efficiency of 90%. Wei et al. [47] proposed an automatic method to extract mature fruit from complex agricultural background getting an accuracy value of 95%; it is based on an improving Otsu threshold-based algorithm using a new feature in OHTA colour space [48].

Choi et al. [49] introduced a new guidance line extraction algorithm to improve the navigation accuracy of weeding robots in paddy fields. The error of the slope angle for the centre of the guidance line is less than 1°, however, it increases gradually with plant size. The rice rows are detected from images acquired with a camera equipped with an IR band-pass filter (> 795 nm). Each image is filtered with a median filter to eliminate noise, after that, the image is processed with using Otsu's method to get the binary image of the rice rows.

Torres et al. [50] proposed an automatic method for vegetation detection in herbaceous crops by a thresholding object based image analysis algorithm through the Otsu method. Images were captured with two sensors (a conventional visible and a multispectral camera) mounted on an unmanned aerial vehicle and acquired over the cornfields, sunflower and wheat. The vegetation classification error ranges between 0% and 10%. Yang et al. [24] detected greenness in images from maize fields achieving an accuracy of 95%. The RGB image was transformed to HSV colour space [51]; from Hue channel (H), the smallest (h_1) and largest (h_2) values were selected. Then, for each pixel; if $h_2 < H(x,y) < h_1 \Rightarrow R(x,y) = G(x,y) = B(x,y) = 0$. The new RGB image is processed with the scheme displayed in Fig. 3.2 Chapter 3 with ExG as the index to obtain the grey image. Their method is sensible to illumination changes. Jiang et al. [22] describes a crop row detection algorithm based on multi-ROIs for agronomic images (wheat, corn and soybean), they reported an accuracy of 93%. The crop row detection is made over the vegetation, once it has been segmented according to the process displayed in Fig. 3.2 and the Expression 3 in Table 2.1.

Meng et al. [59] developed an agricultural system for crop inter-rows weeding. Images were processed in HSI space [60] to detect vegetation (pixels are considered vegetation if their Hue value is between 120 and 160). From green detection the crop rows detection is performed. The average error for crop line detection is below 2.7 cm, with the maximum error value of 6.8 cm. The efficiency depends on the vegetation segmentation however

Abbreviation		Expression
Normalization		$R^* = R/max(R)$, $G^* = G/max(G)$, $B^* = B/max(B)$ $r = R^*/(R^* + G^* + B^*)$, $g = G^*/(R^* + G^* + B^*)$, $b = B^*/(R^* + G^* + B^*)$
Grey		$0.2898 * r + 0.5870 * g + 0.1140 * b$
Gray_1	[22]	$1.262 * g - 0.884 * r - 0.311 * b$
*ExG	[52]	$2 * g - r - b$
ExR	[53]	$1.4 * r - g$
CIVE	[54]	$0.441 * r - 0.811 * g + 0.385 * b - 18.78$
ExGR	[55]	$ExG - ExR$
NDI	[56]	$(g - b)/(g + b)$
GB	[52]	$g - b$
RBI	[57]	$(r - b)/(r + b)$
ERI	[57]	$(r - g) * (r - b)$
EGI	[57]	$(g - r) * (g - b)$
EBI	[57]	$(b - g) * (b - r)$
VEG	[57]	$g * r^a * b^{(a-1)}$
COM	[45]	$0.25 * \text{ExG} + 0.30 * \text{ExGR} + 0.33 * \text{CIVE} + 0.12 * \text{VEG}$
NDVI	[46]	$(NIR - \tilde{R})/(NIR + \tilde{R})$
OTHA	[47]	$r - g$

*Ribero et al. [58] proposed the same colour index obtaining the constant values for each channel by using genetic algorithms.

Table 2.1: Colour channels and colour vegetation indices (CVIs). R, G and B are the components of the RGB image. $\tilde{R}$ and NIR refers to the visible red and near-infrared spectrum respectively.

the segmentation results are not provided. Guijarro et al. [25] introduced a strategy to distinguish between soil and green plants in images from maize fields with an accuracy of 92.09%. The methodology includes: estimation of discrete wavelets, extraction of texture descriptors and a thresholding process. During the thresholding process, pixels with a spectral green value which is 10% higher, relative to the other two components are extracted. Then, a human expert removed false positives manually. Balasubramaniam and Ananthi [61] provided a method to segment nutrient-deficient regions in images from different crops containing missing pixels. They achieved an accuracy value between 92% and 99%, vegetation segmentation results are not shown and their method was based on intuitionistic fuzzy C-means algorithm.

Kazmi et al. [23] presented a methodology to detect thistle in sugar beet fields with an accuracy up to 97%, by using a classifier function constructed with the Mahalanobis distance and linear discriminant analysis. The input data is the combination of various colour vegetation indices, Table 2.1. They have another similar proposal for weed detection in scanned leaf images (outdoor images were not included in the analysis) by using the Bag-Of-Words scheme with K-Nearest Neighbours (KNN) and Support

Vector Machine (SVM) [6]. Again, the input data is the combination of CVIs reported in [23] achieving an overall classification accuracy over 99%.

Ye et al. [62] introduced a probabilistic super-pixel Markov random field method. It was applied for crop detection under strong illumination changes. The method yields the highest mean value of 92.29% with the lowest standard deviation of 4.65%. Cheng et al. [63] proposed a method for weed and rice identification based on image feature analysis and machine learning-based techniques achieving a precision of 98.8%. To classify weed from rice, Harris point finder is first applied to locate interest points, then, multiple features are extracted which are subsequently classified by using a machine learning-based algorithm (decision trees). Moorthy et al. [26] developed a vegetation segmentation method in sugar beet and maize images with an accuracy of 87%. Their method was based on a Naïve Bayesian model using features from RGB and HSV colour spaces. In this work the features used to train the model were manually extracted.

Santos et al. [64] proposed a 3D plant modelling for plant phenotyping by using a hand-held, free moving around the plant, camera to recover a sparse 3D point cloud sampling the plant surface; the samples were processed with spectral clustering to get the 3D plant representation. The methodology was suitable for automatized plant phenotyping facilities or small laboratories. Their experiments with maize plants were unsuccessful.

Ionescu et al. [65] presented a method for biomass type identification in images obtained from mobile devices with accuracy of 90%. For biomass characterization: texture features, local texton dissimilarity and BoW representation, were used. Finally, Montalvo et al. 2016 [27] provided a method based on principal components and CVI metrics to split soil from plants, with an error of 8.07%. The ground-truth images were built semi-automatically by processing the RGB colour channels and with the help of a human expert.

2.1.3 Obstacle detection

Obstacle avoidance is a classical problem in robotics and of course in autonomous vehicles; literature in obstacle detection is vast and well known in image processing. Methodologies reported in literature are diverse (see the references [66, 67] and the references therein). They include techniques based on: optical flow [68, 69], temporal differencing [70], mixture of Gaussians [71], and point detection or changes [72], just to mention a few. Significant progresses have been achieved; however, important improvements are still possible in many cases, due to: poor image acquisition, variations in the lighting conditions at the scene, shadows projected by foreground objects that

could be detected as real objects, multiple objects moving at the scene for both long and short periods and perhaps also occlusions [73]. Object detection tasks increase complexity when they are carried out in video sequences. Many works have been published for object detection on video sequences acquired by static [74] and non-static cameras [75]. Moving camera segmentation is much more challenging than static camera segmentation because two issues should simultaneously be addressed: background motion and the motion of each foreground-moving object. We refer the reader to [76] for a survey on moving object segmentation methods. In particular, for agricultural applications the following works are relevant in the development of the method described in Chapter 4.

Reina and Milella [77] presented a self-learning framework for scene segmentation by an autonomous agricultural tractor with trinocular stereo-vision. Segmentation was based on a self-learning classifier assuming that the vehicle starts its operation from an area free of obstacles. Scenarios analysed include static and non-static obstacles achieving an average classification rate of 91.0%. In contrast to their work, our methodology does not require a training process; obstacles might appear on the scene since the beginning.

Biber et al. [78] incorporated a LIDAR (Light Detection And Ranging) sensor for range finding in their navigation system to detect unexpected objects. In this regard, each tractor in the fleet of RHEA was equipped with a LIDAR system to detect, localize, and classify trees with the aim of safe navigation with favourable assessment [79]. The main advantage of LIDAR sensors is that they are not influenced by sunlight with high infra-red radiation content, which is not fully achieved with Kinect sensors. Despite the effectiveness of these LIDAR-based systems, different researches have demonstrated the outperformance of combined LIDAR and machine vision systems, particularly in dynamic outdoor scenarios [80–82].

In [83], authors developed an obstacle detection system using stereo matching to generate a point cloud distribution at specific regions in the image. Ground plane detection was used to determine the points in the cloud belonging to obstacles. The detector operates on colour spaces; when the colour of the obstacle becomes similar to that of the terrain the obstacle becomes more difficult to detect. Our proposal is capable of working despite this drawback; we add texture information in order to capture obstacles with similar colour to that of the agricultural environments. Texture is also considered in [84] where the information provided by a Kinect with the local binary pattern code was used as a feature to determine if a given pixel is considered as an obstacle or discarded. The algorithm requires a training process to achieve the texture patterns used by the classifier; our method does not require any training process.

Talukder et al. [85] proposed a method based on the height difference and slope between three-dimensional points. Despite having been used successfully, the method has a high computational cost. In 2013, Mendes et al. [86] designed a GPU-based parallel version of the obstacle detection proposed in [85] with the aim of improving the efficiency time. The system performance was conducted using point clouds from a stereo camera and an RGB-D sensor. Similar to Mendes, Wei et al. [87] developed a stereo-based system for people detection with the aim of safe navigation in agricultural environments. An obstacle detection system was recently proposed in [3]. 3D points are sampled into a 2D grid, the relations to the neighbourhood were examined and the obstacles grouped. After this process, a 3D terrain representation with abstraction was generated. Unfortunately, given the architecture of the fleet in the RHEA project we do not have additional data; our information is limited to the video sequences acquired with a monocular camera.

Cheein et al. [88] presented an algorithm for localization and mapping based on the detection of olive stems. They were detected by means of both, a laser range sensor and a monocular vision system. Authors do not consider obstacle detection as a relevant issue. Our method is capable of detecting stems; therefore, our method can also be used for path localization and path mapping tasks. Bayar et al. [89] introduced a new method for increasing the trajectory tracking performance for autonomous orchard vehicles. In order to detect obstacles and trees they used a laser range finder. They achieved good performance to detect trees trunks; unfortunately their results do not include obstacle detection analysis. Gázques et al. [90] have proposed solutions for humans and robots, working together in agricultural environments. The system is a hybrid between autonomous and classical tele-controlled vehicles. The vehicles were equipped with micro programmed intelligence-based systems against the unforeseen, requiring a moderate supervision over the robot by the operator. The automaton was programmed to stop when the sensors detect danger of collision. In contrast with this proposal, we focus our work on developing an automatic process for obstacle detection based on the analysis of image sequences as expressed above.

A robust algorithm that detects a standardized object has been published in [91]. Authors reported a precision of 99.9% in crop rows and 90.8% in grass mowing. The algorithm is based on an existing deep convolutional neural net. However, it is unable to detect people or any different element such as the standardized object. The accuracy of our approach for obstacle detection is not as high as the one reported in [91], but instead, a major advantage is that we do not need any training process, while we also are able to detect elements such as people, trees or vehicles, among others.

As noted throughout this overview, despite the progress reported in the state-of-the-art, there is still work to be done to bring autonomous vehicles, particularly in agricultural

environments, which is a motivational element of our research.

2.2 Machine vision system in the RHEA project

One of the objectives of the RHEA project was to design a whole system for crop/weed identification in maize fields for row following and weed discrimination to apply site-specific treatments. Crop row detection is the base for both weed discrimination and guidance and requires the localization and identification of the crop rows in a region of interest in front of the tractor (rectangular area 3 m wide and 2 m long, Fig. 2.1a).

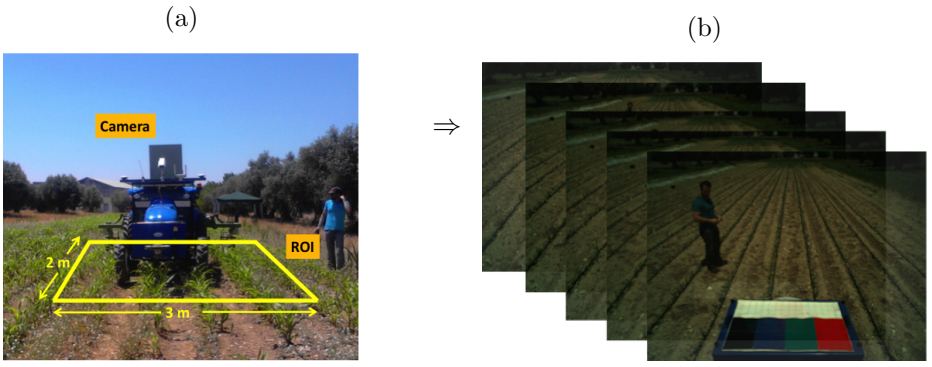


Figure 2.1: Frontal view of the camera pointing to the selected area (SA); b) Consecutive images acquired with the monocular camera mounted on the tractor in a).

The perception system consisted of three main sensors: machine vision with a colour-based camera, Inertial Measurement Unit (IMU), and RTK-GPS (equipped with two antennas: one for XYZ positioning and the other for heading calculations). The camera-based sensor is the SVS4050CFLGEA model from SVS-VISTEK (SVS-VISTEK2014), built with the CCD Kodak KAI 04050M/C sensor with a GR Bayer colour filter; its resolution is 2336 (Horizontal) by 1752 (Vertical) pixels with a 5.5 by 5.5 micrometre pixel size. The digital images were captured with a frequency rate of 2.4s under perspective projection and stored as 24-bit raw colour images in the RGB colour space. The operation speed defined for this application was 0.83 m/s (3Km/h).

The monocular colour-based camera was installed on-board a tractor pointing to a selected area, it was located at 2 m height from the ground and inclined 45° with respect to the ground, Fig. 2.1a displays the frontal view of the camera position. More than 110 image sequences of different length Fig. 2.1b, the longest with 2099 images while the shortest with 20 images (about 15000 frames) were captured during April/May/June 2012 to 2014, including videos during the final demonstration of the RHEA project in May 2014, in a 1.7 ha experimental maize field in La Poveda Research Station, Arganda del Rey, Madrid. All acquisitions were spaced by five/six day periods, that is, they were

obtained under different illumination conditions and different plant growth states for plants and containing people at the working space.

2.2.1 Image pre-processing

Agricultural images are shown to be a potential alternative given their low cost of operation in environmental monitoring, high spatial and temporal resolution, and their high flexibility in image acquisition programming [38]. For example current crop status (including maturity period and identification of disease, pest and weed infestations) and fruit identification for harvesting can be discerned by means of the image analysis improving the decision making for vegetation measurements, irrigation, fruit sorting, etcetera. Correct identification of the elements on the images is an essential step towards the automation of the processes reducing the manual tasks and increasing productivity [30, 31]. Several methods have achieved good performances based on image analysis in agricultural environments; however there are still current limitations and problems coming from the image complexity such as:

a) Uncontrolled illumination: in images coming from outdoor scenarios a variety of weather conditions may appear; highly sunny or cloudy days with different intensities, clear days alternating with different cloud densities, etc., producing shadows on the images coming from plants in a neighbourhood.

b) Distorted images: in agricultural applications the vision system is usually mounted on a mobile device such as in the RHEA tractors. Due to this fact, images are captured with noise effects (multi angle, bidirectional reflectance distribution function and motion) for example. Small movements are detected derived from the motor vibrations, on images captured with cameras mounted on tractors. In this context motion is also produced from the navigation system, tractors moving over rough terrain. These undesired effects can be significant when the exposure time is high.

c) Images with variations in the growing plant states.

d) Dynamic background: in a real time system, the background image content changes continuously. Although sometimes changes are not visually noticeable, however, they must be considered for obstacle detection in order to design an automated system.

e) Large amount of information: efficient image acquisition systems are designed to capture a large number of high-resolution images requiring a set of techniques and technologies able to analyse and process the underlying information.

f) Vignetting effect: this effect is usually generated by the optical system with rays traversing the lens and producing that the image gets darker in the corners of the image. There are several types of vignetting; depending on the optical design of the lenses [92], in the case of images acquired with the vision system of RHEA project, this effect was produced and removed by applying the correction described below [19].

In the RHEA project this effect becomes evident because the Schneider UV/IR 468 cut-off infrared filter used, cuts red wavelengths, which are attenuated causing an excess of greenness on the green channel, becoming more pronounced in the four corners than in the central part of the image. That means that the real colour is displaced toward green and blue at the expense of the read tone, Fig. 2.2a. The process to minimize the vignetting effect in a colour image I of size $H \times W$ is as follows. The first step consists in generating an image pattern with the same dimension as the image, $H \times W$. The pattern contain values ranging in $[0, 1]$, Fig. 2.2b, they are assigned using the centre of the pattern as reference; values in the centre are smaller than values at the extremes. Once the image pattern is computed, the colour channels in I are processed separately, for example, in the red channel. Each pixel is modified with the expression $R(x, y) = R(x, y) + v_r P(x, y) R(x, y)$, where v_r represents the trade-off between corrections in the red channel. The same process is for pixels in the green and blue channels with v_g and v_b values respectively. The result of these transformations is the corrected image free of the vignetting effect, 2.2c. Algorithm 7 Appendix B summarizes the correction process described. Due to the behaviour of the cutting filter, values of v_g and v_b are fixed to 0; this means that only the read channel is corrected. The v_r value was set to 0.3; it had been verified as appropriate for green identification in our image dataset [93, 94].

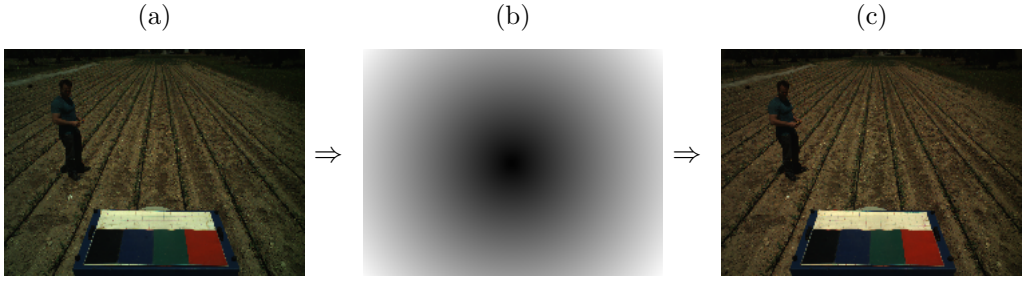


Figure 2.2: a) Image captured with the vision system described in Section 2.2. The greenness excess in the image corners, vignetting effect, can be appreciated; b) Patter correction; c) Image after vignetting removal.

As a final general conclusion, robust methods must be developed to address the issues previously described for the machine vision system before it can be used routinely in farming operations.

Chapter 3

Vegetation segmentation

Vegetation segmentation is a relevant issue in PA tasks, a wide revision of the state-of-the-art in vegetation detection for AI had been introduced throughout Chapter 2, concluding that thresholding algorithms and machine learning strategies are the most commonly used methods for image segmentation. In this chapter, an alternative methodology for vegetation segmentation in images coming from maize fields is presented. It consists of two main steps: a) low-level segmentation and b) class label assignment using Bag of Words representation in conjunction with a supervised learning framework. Several experiments were conducted to evaluate the method efficiency including a texture analysis and a comparison against several well-known existing strategies in current literature. The experimental results have proved the method efficiency to extract green plants with high accuracy in the images used for testing. The research conducted in this chapter was published in [5] and [7].

The chapter is organized as follows: The problem statement, main objectives, and relevant issues related to segmentation are provided in the introduction, Section 3.1. The methodology for vegetation segmentation is described in Section 3.3 followed by the experimental results and comparative analysis in Section 3.4.

3.1 Introduction

In AAVs, vegetation segmentation is a critical step towards the development of different activities in the crop field such as crop row detection, counting plants for germination or monitoring, weed detection for early site specific treatment, or nutrient application, among others, Fig. 3.1. This task is usually carried out from images acquired by the machine vision system on board the AAV and must therefore be considered in the design

of AAVs. In short, a good algorithm to split an image into foreground (maize/plants) and background (soil, irrigation pipes, etc.) is highly demanded to improve the activity performance carried out by the AAV.

Figure 3.1: Some illustrative examples of agriculture images where greenness detection can be used for; weed, obstacle, fruit, pest, and crop row detection

Although several methods for vegetation detection have been already introduced in Chapter 2 many improvements are still possible due to uncontrolled lighting conditions, including sudden shadows, excessive or poor illumination with high variability as discussed previously.

Also, from a literature review it was noted that the commonly used process for green extraction is based on the following main steps [6, 20–25, 44, 47, 49, 59]: i) grey image, the original colour image is transformed into grey scale to enhance the colour channel of interest using expressions in Table 2.1 Chapter 2; ii) thresholding, to separate relevant parts (plants and soil) from the grey image, where Otsu's algorithm is a common automatic method to carry out this task [6, 22, 23, 25, 44, 47, 49, 50]; iii) and finally, the binary image is filtered for noise removal. This process is illustrated in Fig. 3.2.

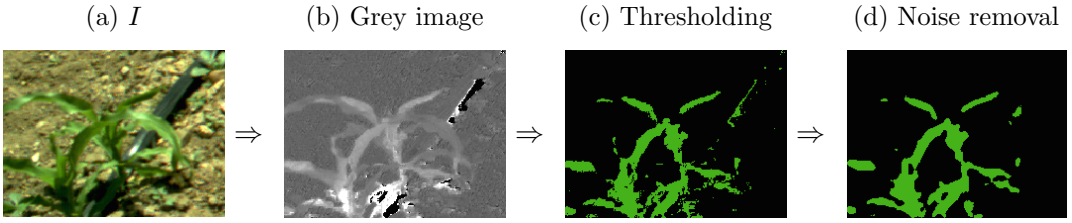


Figure 3.2: Scheme to segment green plants: a) Colour image of size $M \times N$; b) Image in grey scale computed with the Expression 4 in Table 2.1 Chapter 2; c) thresholding of grey image with Otsu algorithm; d) filtered image for noise removal. Images in b, c and d have the same size as the colour image $M \times N$.

Vegetation segmentation in agricultural images is extensively studied with the following main goals:

a) To develop a method for green plant segmentation in images from maize fields with a supervised learning strategy.

b) To reach the above goal, an image dataset of images and the corresponding true-segmented images must be provided. This is the second goal; i.e. the generation of the dataset useful for reference and testing in future research.

Images to form the dataset were selected from images acquired with the machine vision system described in Chapter 2; their true-segmented images are made by inspection and manually touched up as described in Section 3.2. The dataset is subsequently used to design the vegetation segmentation method where a new agricultural image is segmented according to the following steps: a) the original colour image (Fig. 3.2a) is partitioned into several regions of interest (ROIs) Fig. 3.3a; b) feature descriptors are extracted for each ROI (e.g.; colour, texture, geometric structure, etc.) Fig. 3.3b; c) a classifier model is used (Fig. 3.3c), which is previously trained to discriminate between vegetation and background, where the final goal is to determine if the region under analysis is vegetation or not Fig. 3.3d. A detailed description about the method is provided in Section 3.3.

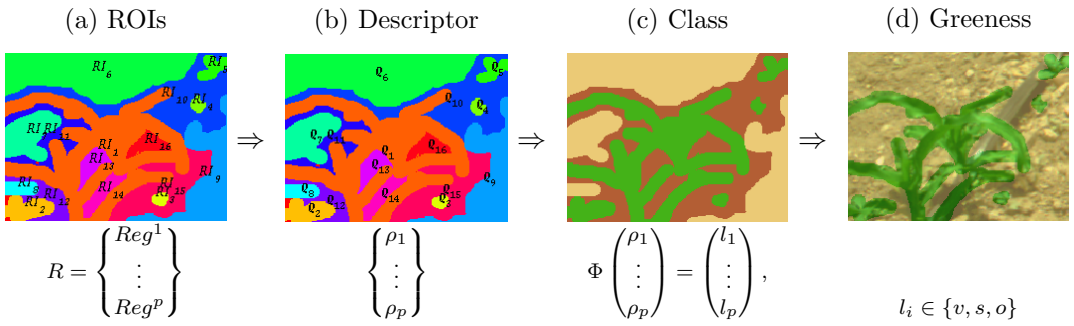


Figure 3.3: Vegetation detection in the colour image in Fig. 3.2a: a) Splitting into $p-$ROIs. b) For each ROI, a feature descriptor is computed. c) The $p-$ descriptors are evaluated by using a classifier model to get a $p-$label, one per region/descriptor. Then, at each pixel in the image the label, associated to the region at which it belongs, is assigned. d) Green pixels detected. All images have the same size as the original colour image in Fig. 3.2a, $M \times N$.

3.2 Image dataset

The method for vegetation detection discussed in this chapter is based on a classifier function used to discriminate between vegetation from the remaining elements on the image. The classifier function is built using a supervised learning strategy and its success depends on the dataset used to train it. If the data is sufficiently representative, the model will work efficiently. To train the model, a set of images and their corresponding true-labelled images are required. Then, from the image dataset available and described in Chapter 2, 168 images were randomly selected for training. Assuming that the vignetting effect was previously removed as explained in Chapter 2, the images and their corresponding labelled images are obtained described as follows.

Study area selection

In the RHEA project, images were acquired with the aim to detect crop rows and weeds in a surface close to the tractor [95]. This work is focused on the area in front of the tractor. Then, from each original colour image, a surface with size $[H \times W] = [450 \times 920]$ pixels is extracted obtaining as a result a new colour sub-image dataset, Fig. 3.4: $\Omega = I_1 \dots I_{168}$, this set will be used to design and assess the segmentation method performance. Therefore, it is mandatory to build another set containing the true-labelled images.

Figure 3.4: Left: Original colour image of size 2336×1752. Centre: Selected area of size 450×920 for vegetation analysis. Right: True-labelled image.

Labelled images

In a labelled image, each pixel is assigned to a single class. In this work, three class labels were considered. An example of a colour image and its labelled image is visible in the second column in Fig. 3.4, with one label per colour. The use of three labels is justified as follows.

In agricultural images, three predominant components are easily identified: green plants, soil and "others". The last class represents the pixels on the border between plants and soil, as well as, pixels belonging to any other element not identified in the image as specific agricultural elements. Under this assumption, each image in Ω was manually labelled by using image-editing software assigning each pixel in the original image to a single class: vegetation (v), soil (s) or others (o). As a result of this handmade procedure each image in Ω has associated its ground truth labelled image, extending this set as follows: $\Omega = \{(I, I^l)_1, \dots, (I, I^l)_{168}\}$. Illustrative examples are shown in Table 3.1, colour images in the first row and their corresponding handmade-labelled in the second row. In the labelled images, the green represents vegetation, the light-brown and the dark-brown represent the soil and the other elements respectively. It is important to remark that the inclusion of the third class is justified by the fact that the manual segmentation on the vegetation borders is even more difficult to carry out under the supervision of an expert. Moreover, we noted that the vegetation detection accuracy using a tri-class classifier (v, s, o) is greater than the accuracy achieved with a bi-class

classifier. This last point is widely discussed in the comparative analysis, Section 3.4. The full set of images is open and available for researchers [96].

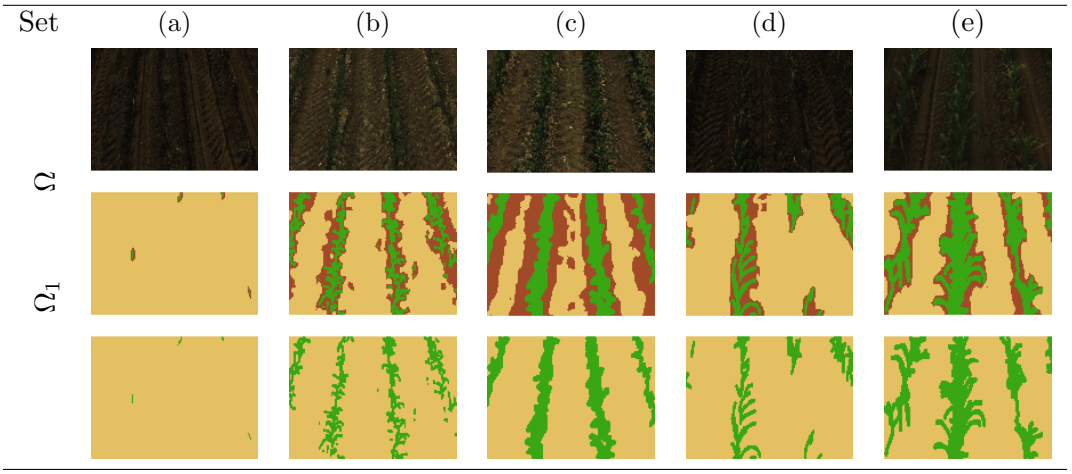


Table 3.1: First row: RGB images. Second row: hand-labelled images (vegetation, soil and other). Third row: Binary images coming from labelled images in second row, foreground (vegetation) and background (joint soil and other).

Dataset partitioning

The 168 images in Ω are partitioned into two complementary sets; $|\Omega_1| = 26$ and $|\Omega_2| = 142$, $(\Omega_1 \cap \Omega_2 = \varnothing)$, the first for building a tri-class classifier (v, s, o) and also for learning the *visual vocabulary* in the BoW representation described in Section 3.3.2, while the second one is used to assess the segmentation method performance.

For the tri-class classifier design, Ω_1 must be again split into training and testing sets in order to avoid data overfitting [97]: $|\Omega_1^A| = 20$ and $|\Omega_1^B| = 6$. In particular, data on these sets will be processed per regions, Fig. 3.5. Instead of working with the full images containing $a = 1005$ regions (346−vegetation, 171−soil and 488−others) in Ω_1^A and $b = 739$ regions (399−vegetation, 166−soil and 174−others) in Ω_1^B, then; $\Omega_1^A = \{(Reg^1, l^1), \ldots, (Reg^a, l^a)\}$ and, $\Omega_1^B = \{(Reg^1, l^1), \ldots, (Reg^b, l^b)\}$, in both cases $|Reg^i| \neq |Reg^j|$ and $l^i \in \{v, s, o\}$.

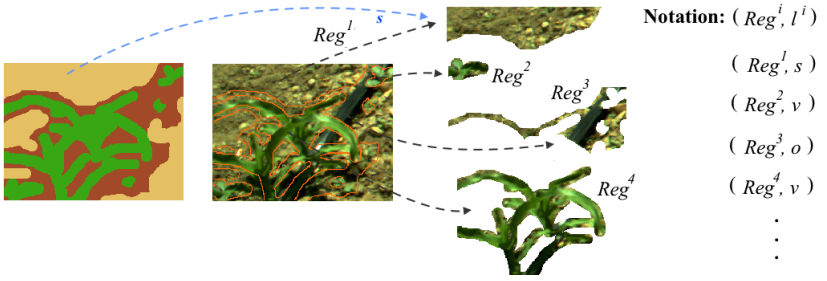


Figure 3.5: Data structure. Regions and labels are processed instead of the full images for the tri-class classifier design.

Now, quality segmentation assessment, labelled images in Ω_2 are first converted into the binary format changing pixels with "s" label to "o" label as expressed in Eq. 3.1 obtaining $\Omega_2 = \{(I, I^b)_1 \dots (I, I^b)_{142}\}$. These binary images are displayed in the third row of Table 3.1.

$$I_{bin}(x, y) = \begin{cases} 1 & \text{if } I_{lab}(x, y) = v, \\ 0 & \text{otherwise.} \end{cases} \tag{3.1}$$

Once the colour image I is processed with Algorithm 1 (Section 3.3.4) the resulting segmented image I_{bin} is compared against the truth binary image I^b. Then, the accuracy value is computed by counting the number of matching labelled pixels between I_{bin} and I^b. In order to objectively measure the overall quality of the segmentation, several popular metrics were considered [4]. They are listed in Table 3.2.

ID	Description	Expression
n_{TP}	True positives	Pixels in I_{bin} with label v whose real label in I^b is v
n_{TN}	True negatives	Pixels in I_{bin} with label v whose real label in I^b is o
n_{FP}	False positives	Pixels in I_{bin} with label o whose real label in I^b is o
n_{FN}	False negative	Pixels in I_{bin} with label o whose real label in I^b is v
OSR	Overall success rate	$(n_{TP} + n_{TN})/(n_{TP} + n_{TN} + n_{FP} + n_{FN})$
TPR	True positive rate	$n_{TP}/(n_{TP} + n_{FN})$
TNR	True negative rate	$n_{TN}/(n_{TN} + n_{FP})$
PPV	Positive predictive value	$n_{TP}/(n_{TP} + n_{FP})$
NPV	Negative predictive value	$n_{TN}/(n_{TN} + n_{FN})$
F	F-measure	$(2 * n_{TP})/(2 * n_{TP} + n_{FN} + n_{FP})$

Table 3.2: Statistical measures for performance evaluation to compare a segmented image I_{bin} with its truth labelled binary image I^b [4].

3.3 Methodology description

Our aim is the partition of an agricultural image into three non-overlapped regions, vegetation, soil and others, so that each region is homogeneous and the union of two disjoint regions is homogeneous [98]. An example of a segmented agricultural image is displayed in Fig. 3.3c, details of the full segmentation process are portrayed in Fig. 3.3. ROIs extraction, feature descriptors and class association are described in this section.

3.3.1 Region of interest extraction

Without knowledge of the image structure the first step is to find nearly uniform regions – ROIs. The principle is that pixels in small regions tend to contain elements of the

same class, Fig. 3.3a. Ideally, each ROI should contain a single class of elements; v, s or o. Then, each pixel in an original colour image $I_i \in \Omega$ is assigned to a unique region: $R_i = \{Reg^1, \ldots, Reg^p\}$,$|R_i| = p$, where the $j-$th element contains the pixels into the $j-$th ROI: $Reg^j = \{(x,y)_1, (x,y)_2, \ldots, (x,y)_m\}$, $|Reg^j| = m$, $(x,y)_t \in Reg^j$ contains the pixels spatial positions in image I_i. Note, the number of pixels can be different for each region, meaning that $|Reg^1| \neq |Reg^2| \neq |Reg^p|$.

Different methods have been proposed in literature to get ROIs. In this research the more relevant, from the point of view of agricultural image segmentation, were considered: K-means (KM) [99], Self-Organization Maps (SOM) [100], Fuzzy C-Means (FCM) [101] and Over-Segmentation (OS) with a graph-based method [102]. A short description of these partitioning methods is provided in Appendix A.

3.3.2 Feature descriptors

Once an image is split into several regions the next step is to find an appropriate set of features to characterize the pixels belonging to each region. The features will be used in subsequent steps to determine the regions belonging to vegetation. The question is: which information should be used to represent the regions? (Fig. 3.6b). To answer this question, several well-known local properties existing in literature are analysed in order to find the best to describe pixels belonging to vegetation. Then, the aim of this section is to identify a single feature descriptor to represent each region. For simplicity and illustrative purposes, the $j-$th ROI displayed in Fig. 3.3a - orange region with m pixels - is used in the subsequent paragraphs for feature extraction.

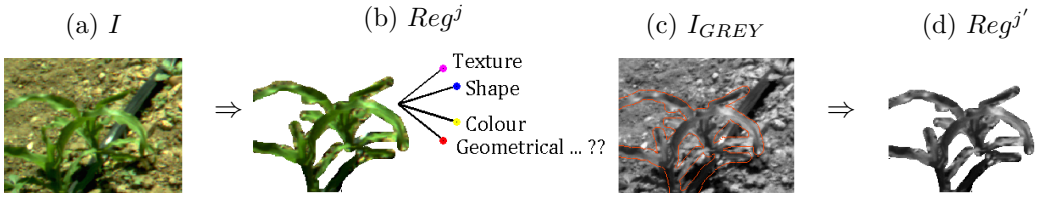


Figure 3.6: a) Colour image, size $M \times N$; b) Colour pixels belonging to the orange region in Fig. 3.3a; c) Grey scale image, size $M \times N$; d) Pixels in b) in grey scale.

In the above figure, $Reg^{j'}$ denotes pixels in grey scale belonging to the region under analysis, Fig. 3.6d. This is because some features are computed from the grey scale image, which is previously transformed with any vegetation index, such as the ones provided in Table 2.1.

Texture analysis

Quantification of texture descriptors is computed based on the statistical analysis in the grey scale space. It is characterized by the repetition of basic elements providing information about the spatial arrangement of the intensities in the image/ROI.

First order statistics (ρ_{fs}): computed from the histogram of the intensity values in $Reg^{j'}$. The histogram is a summary of the statistical information into the ROI, which is obtained by counting the number of pixels at each different intensity level and divided by m (number of pixels in the ROI) to get a normalized vector, Fig. 3.7. From the counts vector, six metrics are computed and joined to obtain a descriptor of dimension $z = 6$. Metric expressions are provided in Table B.1 Appendix B.

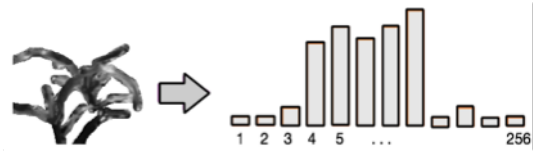


Figure 3.7: Histogram of the intensity values in the ROI.

Autocorrelation (ρ_{ac}): the autocorrelation function of a grey scale image, size $M \times N$, for displacement (d_1, d_2) where $d_1 \in [0, M]$ is a displacement in rows (downward) and $d_2 \in [0, N]$ is a displacement in columns (to the right), is given by Eq. 3.2 [60]. In this work, the descriptor consists of the second order statistics of the autocorrelation coefficients belonging to the region $Reg^{j'}$; mean and standard deviation are considered having a two dimensional descriptor $z = 2$.

$$c_{x,y,d_1,d_2} = \frac{\frac{1}{(M-x)(N-y)} \sum_{i=1}^{M-x} \sum_{j=1}^{N-x} I_{GREY}(i,j) I_{GREY}(i+d_1, j+d_2)}{\frac{1}{M \times N} \sum_{i=1}^{M-x} \sum_{j=1}^{N-x} I^2_{GREY}(i,j)} \tag{3.2}$$

Grey level run length matrix (ρ_{rlcm}): a run-length matrix $p(\hat{i}, \hat{j})$ is defined as the number of runs with pixels of grey level $\hat{i}$ and a run length $\hat{j}$ [10]. A $p-$matrix is extracted from $Reg^{j'}$ and metrics in Table B.2 Appendix B, are computed and joined to get a single descriptor with dimension $z = 11$.

Grey level co-occurrence matrix (ρ_{glcm}): is a 256×256 matrix $GCM(\hat{i}, \hat{j})$, where each element contains the number of pairs of pixels with intensities $\hat{i}$ and $\hat{j}$ into $Reg^{j'}$ [103]. The GCM matrix was computed in four directions obtaining as a result four co-occurrence matrices; GCM^{0°, GCM^{45°, GCM^{90° and GCM^{135°. For each matrix, metrics in Table B.3 Appendix B, are computed to get a vector of dimension $z = 7$. The

average of the four vectors corresponding to the four co-occurrence matrices is the final descriptor.

Mix of descriptors: ρ_{fs}, ρ_{ac}, ρ_{rlcm} and ρ_{glcm} are combined to obtain eleven new elements, expressions from ρ_{mix1} to ρ_{mix11} in Table 3.3. For example; ρ_{mix1} is built by joining items in ρ_{fs} and ρ_{ac} with respective dimensions of 6 and 2, resulting in an $8-$dimensional vector.

Table 3.3 summarizes all descriptors defined above; each descriptor is normalized by dividing each component by the descriptor's magnitude.

Descriptor	z	Descriptor	z
ρ_{fs}	6	$\rho_{mix5} = \{\rho_{fs}, \rho_{rlcm}, \rho_{glcm}\}$	24
ρ_{ac}	2	$\rho_{mix6} = \{\rho_{fs}, \rho_{glcm}\}$	13
ρ_{rlcm}	11	$\rho_{mix7} = \{\rho_{ac}, \rho_{rlcm}\}$	13
ρ_{glcm}	7	$\rho_{mix8} = \{\rho_{ac}, \rho_{glcm}\}$	9
$\rho_{mix1} = \{\rho_{fs}, \rho_{ac}\}$	8	$\rho_{mix9} = \{\rho_{ac}, \rho_{rlcm}, \rho_{glcm}\}$	20
$\rho_{mix2} = \{\rho_{fs}, \rho_{ac}, \rho_{rlcm}\}$	19	$\rho_{mix10} = \{\rho_{rlcm}, \rho_{glcm}\}$	18
$\rho_{mix3} = \{\rho_{fs}, \rho_{ac}, \rho_{glcm}\}$	15	$\rho_{mix11} = \{\rho_{fs}, \rho_{ac}, \rho_{rlcm}, \rho_{glcm}\}$	26
$\rho_{mix4} = \{\rho_{fs}, \rho_{rlcm}\}$	17		

Table 3.3: Descriptors for texture analysis; z is the vector dimensionality. The mathematical expressions to get the descriptors from 1 to 4 rows, first column, are provided in Appendix B. Table descriptors reported in [5].

SIFT-SURF descriptor

There are two well-known descriptors used in computer vision applications because of their good performance in different classification and recognition tasks. For each pixel $(x, y)_t \in Reg^{j'}$ a local descriptor is extracted.

Scale-Invariant Feature Transform $(SIFT, \rho_{sift})$: a 16×16 neighbourhood around the point $(x, y)_t$ is defined. The region defined by the neighbourhood is partitioned into sub-regions of 4×4 size. For each sub-region an orientation histogram with 8 bins is built with each bin covering 45 degrees from $0°$ to $360°$. The SIFT descriptor then becomes a vector containing all values of these histograms resulting in a vector of dimension $z = 4 \times 4 \times 8 = 128$. This vector is normalized to unit length in order to enhance invariance to deal with changes in illumination. The non-linear illumination effects are reduced applying a threshold of 0.2 and is normalized again [104]. The region is represented by $m-$descriptors of dimension 128, a descriptor per pixel into the region.

Speeded-Up Robust Features $(SURF, \rho^{64}_{surf}, \rho^{128}_{surf})$: a 20×20 square region, centred at pixel $(x, y)_t$, is considered. The region is split up regularly into 4×4 square sub-regions. For each sub-region, horizontal and vertical wavelets, d_x and d_y, responses are obtained

and a vector of length $z = 4 \times 4 \times 4 = 64$ is formed, $\rho^{64}_{surf} = [\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|]$. The extended descriptor is included in the analysis ρ^{128}_{surf}. It is a composition of the base descriptor ρ^{64}_{surf} and a couple of similar features using the same sums in ρ^{64}_{surf}, but now these values are split up further. The sums of d_x and $|d_x|$ are computed separately for $d_y < 0$ and $d_y \geq 0$. Similarly, the sums of d_y and $|d_y|$ are split up according to the sign of d_x, thereby duplicating the number of features [105]. Similarly to SIFT descriptor, the region is represented by $m-$descriptors of dimension 64 or dimension 128 depending on the election.

For practical issues from the m-descriptors only a set of Ns-elements, randomly chosen, will represent each ROI.

Colour descriptors

Several previous studies have concluded that vegetation indices accentuate vegetation in the remotely sensed images. In this work, four different combinations of CVIs were used, they were proposed by Kazim et. al [6]. The composition of each descriptor is given in Table 3.4. For example, to get the ρ_{cvi4} descriptor, the colour image is first converted to grey scale with ExG, CIVE, GB, and ERI indices, Table 2.1, obtaining four grey images, Fig. 3.8. Then, each pixel in the region $Reg^{j'}$ will be represented by a four-dimensional vector containing the different grey intensity values. Similarly to the local SIFT and SURF descriptors, from the $m-$vectors a set of $Ns-$elements, randomly chosen, represents the region.

ID	Composition of colour descriptors
ρ_{cvi2}	ExG, GB
ρ_{cvi4}	ExG, CIVE, GB, ERI
ρ_{cvi9}	ExR, ExGR, NDI, GB, RBI, ERI, EGI, r, g
ρ_{cvi14}	ExG, CIVE, ExR, ExGR, NDI, GB, RBI, ERI, EGI, EBI, r, g, b, Grey

Table 3.4: Composition of colour descriptors proposed by Kazim et. al [6].See Table 2.1 for CVIs definitions.

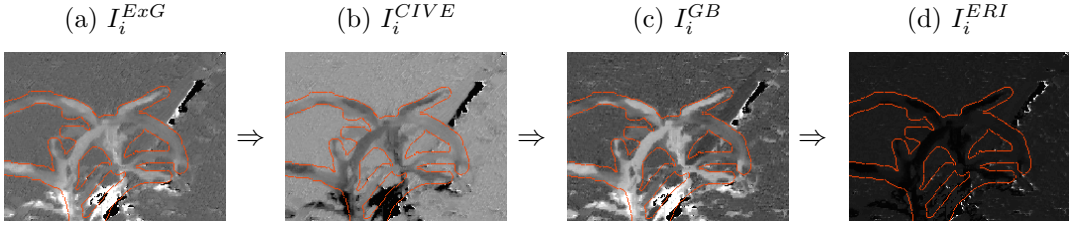


Figure 3.8: Image in grey scale obtained with different CVIs; ExG, CIVE, GB, and ERI, defined in Table 2.1. The orange line encloses the pixels belonging to a single interest region $Reg^{j'}$.

Bag of Words representation

The BoW representation is one of the most popular methods for text analysis [106]. In general the success of this representation is based on the high discriminative power of some words and the redundancy of language. Subsequently, this technique was adapted for computer vision applications [15, 23, 65, 107] where the key idea is to quantize each extracted feature from an image/ROI into one set of visual words, and then to represent each ROI by a histogram of the visual words, usually referred as codebook, Figure 3.9.

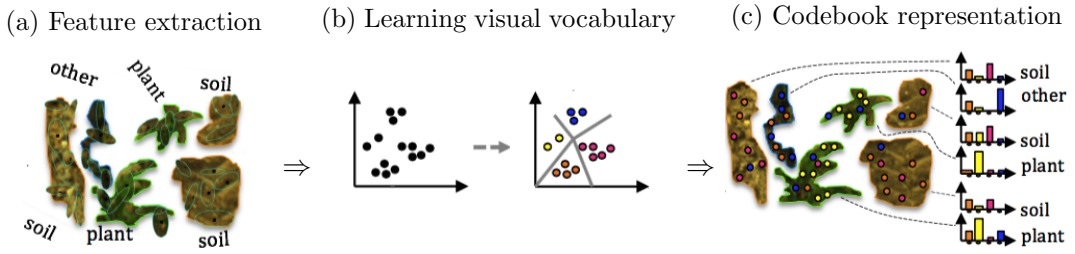


Figure 3.9: Process to learn a visual vocabulary: Feature extraction from the different classes of elements. b) Reduction of dimensionality, usually by clustering methods. c) Each feature is assigned the label of the nearest centre, then in each region the count vector is computed.

To get a codebook, the first step is to train a visual vocabulary often derived by clustering. Then, given a set of ROIs containing the different involved classes (plant, soil and others), local features are extracted, one per pixel for each ROI, Figure 3.9a. All features are pooled and the cluster centres represent the set of visual words, Figure 3.9b. Then, a label is assigned to each ROI based on a minimum distance criterion between the feature describing the ROI and a cluster. Once all ROIs are labelled the frequency of each label is computed, the histogram is the ROI codebook, Figure 3.9c. In this work, the learning process to get the visual vocabulary is computed from regions defined in $\Omega_1^A : R_1^A = \{Reg^1, \ldots, Reg^a\}$. Details of this process are given below.

Step 1. Feature extraction:

Consider a region $Reg^i \in R_1^A$ with $m = |Reg^i|$ pixels. For each pixel in Reg^i a local feature is computed, $F^i = \{\rho_1, \ldots, \rho_m\}, \rho_j \in R^z$, z is the descriptor dimensionality. The same applies for all elements in R_1^A obtaining as a result a set of descriptors $F^A = \{F^1, \ldots, F^a\}$. It is worth mention, the most common feature used in literature is the SIFT descriptor [108]. However, for characterization of vegetation the SURF and colour descriptors in Table 3.4 are also considered in the experimental analysis carried out in this research.

Step 2. Visual dictionary:

This is obtained with the K-means clustering method because it is the most common way of constructing visual vocabularies. All descriptors in F^A are processed to get $K-$centres. Each centre is a visual word $w_i \in \Re^z$. Note that each visual word has the same dimensionality as the local descriptors in F^A. The set of $K-$visual words is the visual dictionary: $W = \{w_1, \ldots, w_K\}$.

Step 3. Codebook representation:

W is used to obtain the codebook for each region in R_1^A. Again, consider the region $Reg^i \in R_1^A$ with $m = |Reg^i|$ pixels and its corresponding set of descriptors $F^i = \{\rho_1, \ldots, \rho_m\}$. Then, a label is assigned to each descriptor with the nearest visual word in the vocabulary, Eq. 3.3, obtaining $\tilde{W}^i = \{\tilde{w}_1, \tilde{w}_2, \ldots, \tilde{w}_m\}, \tilde{w}_j \in W$.

$$\tilde{w}_j = \underset{k}{\operatorname{argmin}} \parallel \rho_i - w_k \parallel^2, j \in [1, K] \tag{3.3}$$

From $\tilde{W}^i$, the frequency of each visual word is computed; $\{c_1 \ldots c_K\}$, c_k is the number of occurrences of the visual word $w_k \in \tilde{W}^i$, defined by Eq. 3.4, where δ is the Kronecker delta function Eq. 3.5. Finally, the codebook for Reg^i is the normalized counts vector: $H^i = \{c_1/m, \ldots, c_k/m\}, H^i \in \Re^z$ with $z = K$. The codebook process is illustrated in Fig. 3.10.

$$c_k = \sum_{i=1}^{m} \delta(\tilde{w}^j - w_k) \tag{3.4}$$

$$\delta(\tilde{w}^j - w_k) = \begin{cases} 1 & \text{if } \tilde{w}^j = w_k \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$

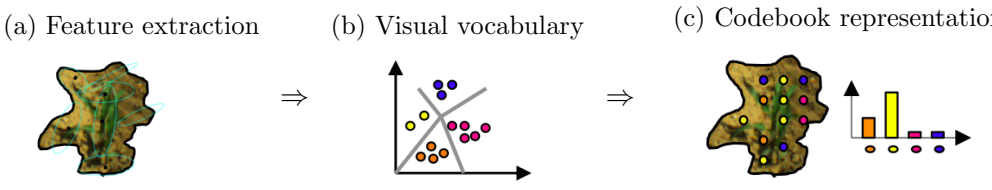


Figure 3.10: Process to get the codebook of a ROI. a) Region of interest, the black points represent the local features. b) Visual vocabulary previously trained Fig 3.9b, each colour represents a visual word. c) Label association at each feature followed by its frequency histogram.

Then, the codebooks for elements in R_1^A are $\{H^1, \ldots, H^a\}$. From now on, the following notation is introduced:

$\Omega_1^A = \{(Reg^1, l^1), \ldots, (Reg^a, l^a)\}, l^i \in \{v, s, o\} \rightarrow \tilde{\Omega}_1^A = \{(H^1, l^1), \ldots, (H^a, l^a)\}$, where $\tilde{\Omega}_1^A$ has the same number of elements as Ω_1^A, $|\tilde{\Omega}_1^A| = a$.

To conclude, H_{SIFT}, H_{SURF}, H_{COM}, H_{CVI2}, H_{CVI4}, H_{CVI9} and H_{CVI14} will be used from now on to distinguish the codebook obtained with the different local features.

3.3.3 Class label assignment

Once a ROI is represented by a feature descriptor, such as those mentioned above, the next step is to determine the class it belongs to, vegetation, soil or others. For this task a SVM classifier is used to get a tri-class classifier. The SVM framework consists of two stages: training and testing. In the first, the tri-class model is built with features extracted from the ROIs, then the model is used to predict the label of a new ROI never seen before into the second stage. The process is illustrated by using the BoW representation, Fig. 3.11.

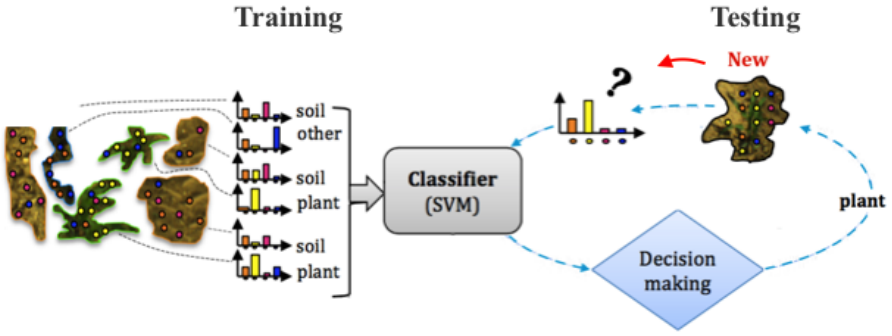


Figure 3.11: Classifier function.

Training: consider the labels and codebooks in $\tilde{\Omega}_1^A = (H^1, l^1), \ldots, (H^a, l^a), l^i \in \{v, s, o\}$. The tri-class SVM model is computed using the one-against-one method. This method builds three bi-class classifiers; (p vs. s), (p vs. o) and (s vs. o), each is trained from data contained in two classes. For training data, with the $i-$th and $j-$th classes, the binary classification problem defined in Eq. 3.6, must be solved. The full minimization process is defined in Hsu and Lin [109]. According to the model in 3.6, the training vectors $H^i \in \tilde{\Omega}_1^A$ are mapped into a higher dimensional space by the function Φ. C is the penalization parameter; it tells the SVM optimization how many misclassifications are allowed at each training.

$$\begin{aligned} \underset{\omega^{ij},b^{ij},\xi^{ij}}{\text{minimize}} \quad & \Psi = \frac{1}{2}(w^{ij})^T\omega^{ij} + C\sum_t \xi_t^{ij} \\ \text{subject to} \quad & (w^{ij})^T\Phi(H^t) + b^{ij} \geq 1 - \xi_t^{ij}, \text{ if } l^t = i, \\ & (w^{ij})^T\Phi(H^t) + b^{ij} \leq -1 + \xi_t^{ij}, \text{ if } l^t \neq j, \\ & \xi_t^{ij} \geq 0. \end{aligned} \tag{3.6}$$

In short, the SVM searches for a balance between the regularization term $\frac{1}{2}(w^{ij})^T\omega^{ij}$ and the training errors. Furthermore, $\mathbf{K}(H^i, H^j) \equiv \Phi(H^i)^T\Phi(H^j)$ is called the kernel function. Different Kernels have been proposed in literature [110]. A linear and a radial basis functions were selected for evaluation, Eq. 3.7 and Eq. 3.8.

$$\boldsymbol{K_L}(H^i, H^j) = H^i(H^j)^T \tag{3.7}$$

$$\boldsymbol{K_G}(H^i, H^j) = exp(-\Gamma \parallel H^i - H^j \parallel^2), \Gamma > 0 \tag{3.8}$$

Testing: once Eq. 3.6 was solved, the performance of the model must be evaluated; the assessment is made with items in set Ω_1^B. After the codebooks are computed, as described in Step 3.3.2 in the BoW representation, the following set is obtained: $\tilde{\Omega}_1^B = \{(H^1, l^1), \ldots, (H^b, l^b)\}, l^i \in \{v, s, o\}$ and $|\tilde{\Omega}_1^B| = b$. Descriptors in $\tilde{\Omega}_1^B$ are evaluated using the following voting strategy: if $sign((w^{ij})^T\Phi(H)+b^{ij})$ says H is in the $i-$th class, then the vote for the $i-$th class is increased by one, otherwise, the $j-$th is increased by one. Then, H chooses the class with the largest vote. When two classes have identical votes, the one with the smaller index is selected [111]. Once all descriptors are evaluated the set of predicted labels associated to each descriptor is obtained; $\hat{L} = \{\hat{l}^1, \ldots, \hat{l}^b\}, \hat{l}^i \in \{v, s, o\}$. Then, the model accuracy is computed by comparing the predicted labels ($\hat{L}$) with the real labels $L = \{l^1, \ldots, l^b\}$ in $\tilde{\Omega}_1^B$ and expression Eq. 3.9.

$$Accuracy = 100 \times \frac{\#\text{ elements correctly labelled in } \hat{L}}{b} \tag{3.9}$$

3.3.4 Vegetation identification

The methodologies described above are the basis to design the vegetation segmentation method, which is summarized in Algorithm 1. The input is a colour agricultural image, I, and a classifier model with a decision function. The output is the segmented image.

The process starts with the image partition, then a descriptor is assigned to each region. Descriptors are evaluated with the classifier decision function to get the label of each

Algorithm 1: Vegetation segmentation, Fig. 3.3.

Input : I image, Ψ classifier function
Output: I_{bin}, 1: foreground/vegetation, 0: background

1 Detection of ROIs in I : $\hat{R} = \{R_1, \ldots, R_P\}$;
2 Descriptor per region : $\Pi = \{\rho_1, \rho_2, \ldots, \rho_p\}, \rho_i \in \Re^z$;
3 Label assignment : $\Psi(\Pi) = \{l_1, \ldots, l_p\}, l_i \in \{v, s, o\}$;
4 Pixel labelling : $I_{bin}(x,y) = \begin{cases} 1 & \text{if } l_i = v \Leftrightarrow I(x,y) \in R_i \\ 0 & \text{otherwise} \end{cases}$
5 Noise removal : $I_{bin} = I_{bin} \bigotimes f_{5\times5}$;

region. Then, a label is assigned to each pixel with the label of the region it belongs to. Finally, pixels with labels s, o are considered background, and pixels with the label v conform the foreground. The resulting binary image is filtered for noise removal and misclassifications by using a median filter with size 5×5.

3.4 Experimental results

Experiments conducted to show the method performance against other existing strategies are described and discussed in this section. Given the image collection of 142 images in Ω_2, the goal is to segment automatically the images with Algorithm 1. To understand how this algorithm works under different image conditions and different parameter values an extensively experimental analysis is carried out. The ranges of parameter values involved in the whole process are summarized in Table 3.5.

Definition	Abbr.	Value
Dataset size, colour images and their labelled images	Ω	168
Classifier model, 70% of elements in Ω	Ω_1	26
Segmentation evaluation, 30% of elements in Ω	Ω_2	142
Number of pixels/descriptors per region	Ns	80
Visual vocabulary size	K	$[50, 2000]$, steps of 50
Penalty parameter in the SVM model, Eq. 3.6	C	$[0.1, 20]$, steps of 0.1
Kernel parameter, Eq. 3.8	Γ	$[0.1, 20]$, steps of 0.1
KM, partitioning method, Section 3.3.1		$\{5, 10, 20, 30, 40, 50, 60\}$
FCM, partitioning method, Section 3.3.1		$\{5, 10, 20, 30, 40, 50, 60\}$
SOM, partitioning method, Section 3.3.1		
Row vector of dimension sizes		$[8, 8]$
Number of training steps		100
Initial neighbourhood size		3
Topology function		hexagonal layer
Distances between the layer's neurons		link distance
OS, partitioning method, section 3.3.1	$(\hat{k}, \sigma', min)$	$(0.1, 300, 100)$

Table 3.5: Parameters involved in the whole process.

The results were computed using the Image Processing Toolbox MATLAB 2016b for 64 bits under Windows 10 and Intel Core 2 CPU, 3 GHz, 4 GB RAM.

3.4.1 Feature selection

An appropriate way to select the descriptor consists of the following steps: a) study the data properties, b) select a suitable classification algorithm, and c) chose the descriptor with optimal performance. However, during the state-of-the-art review we have shown that the agricultural images contain a high level of complexity and are also difficult to analyse. Therefore, in order to limit the dependency of different parameters, a linear kernel function, Eq. 3.8 for the SVM was selected at a first instance to determine the most appropriate feature descriptor for vegetation characterization. It was chosen because it is simple and easy to handle, also because of its good reported performance in agricultural contexts and other environments. The library for SVM provided by Chang and Lin [112] was used to design all classifier models in this research.

Now, to build the tri-class classifier, the correction parameter C must be also estimated, Eq. 3.6. For this, many linear tri-class classifiers with different parameter values, row 6 Table 3.5, were built with elements in Ω_1^A as suggested in Chang and Lin [112]. Then data in Ω_1^B was evaluated with the different generated models and the accuracy value was computed from the number of patterns correctly or incorrectly predicted with the expression in Eq. 3.9. In the case of the BoW representation, the visual vocabulary size K must also be selected. The estimation consisted in testing with different pairs (C, K). The range of values are defined in rows 5 and 6, Table 3.5. Fig. 3.12 displays the performances for different visual vocabulary sizes evaluated for a SVM with the linear kernel Eq. 3.7 and the different BoW descriptors. The best performance is achieved with the pair $(C, K) = (17.2, 1950)$ and H_{COM} as feature descriptor. The library for SIFT descriptors is the one provided by Vedaldi and Fulkerson [113].

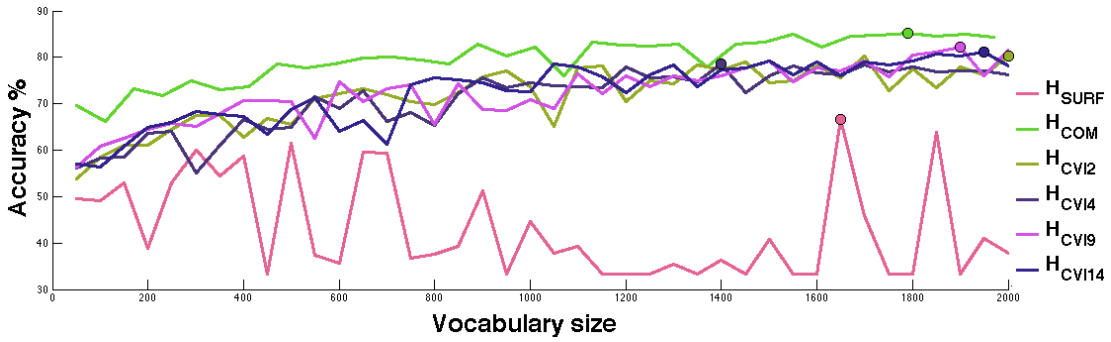


Figure 3.12: Estimation of the vocabulary size for values in Table 3.6, rows 24-30. The H_{COM} descriptor with a vocabulary size of 1790 and SVM with the linear kernel Eq. 3.7 and parameter $C = 17.2$ achieves the best performance are of 85.17%.

Finally, the classifiers with the highest performance are displayed in Table 3.6. The best performance was achieved with the BoW representation which was above 80% for H_{COM}, H_{CVI2}, H_{CVI9}, and H_{CVI14}. These findings suggested that texture, SIFT-SURF and colour descriptors are insufficient for vegetation discrimination in the tested images. From this table it is also easy to verify that the BoW descriptors represent the best strategy. Because of this, new tri-class classifiers were built changing the kernel function for a radial basis function with parameter Γ, Eq. 3.8. The performance results are provided in Table 3.7. In this case three parameters must be adjusted (K, C, Γ). Searching consists in testing with triplet (K, C, Γ) and the one with the best cross-validation accuracy is finally selected.

Results in rows 1 to 16 are similar to those reported in [5], different accuracy rates are obtained due to the image data sets not being the same.

Desc.	z	C	Acc.(%)	Desc.	z	C	Acc.(%)
Texture				Colour			
1.ρ_{fs}	6	16.1	36.350	19.ρ_{com}	1	1.5	44.865
2.ρ_{ac}	2	0.1	38.712	20.ρ_{cvi2}	2	0.1	33.333
3.ρ_{rlcm}	11	2.1	38.683	21.ρ_{cvi4}	4	2.9	50.814
4.ρ_{glcm}	7	18.1	34.925	22.ρ_{cvi9}	9	1.3	48.838
5ρ_{mix1}	8	20.1	37.337	23.ρ_{cvi14}	14	1.3	48.721
6.ρ_{mix2}	19	12.1	42.266	Bag of Words			
7.ρ_{mix3}	15	10.1	40.994	24.H_{SIFT}	128	21.6(1550)	68.12
8.ρ_{mix4}	17	20.1	39.521	25.H_{SURF}	64	21.1(1650)	66.68
9.ρ_{mix5}	24	16.1	40.243	26.H_{COM}	1	16.6(1790)	85.17
10.ρ_{mix6}	13	10.1	37.373	27.H_{CVI2}	2	13.6(2000)	80.25
11.ρ_{mix7}	7	12.1	41.071	28.H_{CVI4}	4	9.6(1400)	78.49
12.ρ_{mix8}	9	0.1	37.416	29.H_{CVI9}	9	17.6(1900)	82.31
13.ρ_{mix9}	20	0.1	40.928	30.H_{CVI14}	14	17.1(1950)	81.20
14.ρ_{mix10}	18	2.1	38.482				
15.ρ_{mix11}	26	18.1	41.588				
SIFT-SURF							
16.ρ_{sift}	128	0.1	51.187				
17.ρ^{64}_{surf}	64	2.5	51.309				
18.ρ^{128}_{surf}	128	0.9	51.727				

Table 3.6: Performance of a tri-class linear classifier model (p, s, o) designed with a linear SVM function using different descriptors of dimensionality z and set Ω_1^A. The table also contains the SVM parameter value C and the vocabulary size W for the BoW representation enclosed in brackets next to the C value.

From Table 3.7, the best values were achieved with descriptors proposed by Kazim et al. [6]. They reported an accuracy of 97% with H_{CVI14} to detect creeping thistle. A similar performance is obtained in the tested maize images used in our experiments; 95.31% with H_{CVI4} and $(K, C, \Gamma) = (1970, 21.6, 20.6)$. This classifier, Table 3.7 row 6, with the highest performance is selected and used in the vegetation segmentation algorithm.

Descriptor	z	(K, C, Γ)	% Accuracy
H_{SIFT}	128	$(1650, 19.1, 21.6)$	90.99
H_{SURF}	64	$(1950, 18.1, 18.1)$	90.38
H_{COM}	1	$(\ 590, 21.1, 6.6\ )$	91.50
H_{CVI2}	2	$(1490, 19.6, 21.1)$	93.83
H_{CVI4}	4	$(1970, 21.6, 20.6)$	95.31
H_{CVI9}	9	$(1670, 14.1, 21.6)$	94.65
H_{CVI14}	14	$(1490, 17.1, 20.6)$	94.84

Table 3.7: Accuracy in percentage of tri-class classifiers designed with SVM and a radial base function as kernel, last column. In the first column, the feature descriptor followed by its dimensionality value. The parameter values are in the third column. Table Results reported in [7].

3.4.2 Partition selection method

The partition selection method is a key step in the vegetation segmentation algorithm. If the region size is too large, the probability that the region contains different classes is high. On the other hand, if the region size is too small, the capture of sufficient information becomes difficult. For this selection, 50 images were randomly selected from Ω_1, the quality of the segmented images was used as the criterion to select the partitioning method, Section 3.3.1. Quality is determined in terms of metrics in Table 3.2.

Again, for practical reasons, the evaluation was made by means of a SVM classifier with a linear kernel and H_{COM} as feature descriptor, achieving the best performance in Table 3.6. Results are displayed in Table 3.8. The average values of OSR and TPR (Table 3.2) are similar in all cases. For each method; KM, COM, FCM, and OS in Section 3.3.1, several parameter values were tested; the range of values are provided in Table 3.5, selection was made following the same quality criterion as above to get the parameter's values in Table 3.8 .

Case	Measure	KM	SOM	FCM	OS
Average	OSR(%)	86.3	86.45	81.47	85.9
	TPR(%)	65.47	68.79	35.92	62.59
Best	OSR(%)	91.1	88.1	86.7	87.8
	TPR(%)	89.3	96.7	84.7	77.2
Worst	OSR(%)	74.8	80.8	61.1	68
	TPR(%)	50	60.9	37.61	42.8

Table 3.8: Evaluation of different partitioning methods using a linear SVM classifier with H_{COM} as feature descriptor. The parameter values were chosen from values in Table 3.5. The OSR and TPR metrics (Table 3.2) are used as performance criterion. Results reported in [7].

From Table 3.8, KM achieves good performance with 30 clusters, while FCM works better with 10. For SOM, Matlab default parameters were used, Table 3.5. In the

case of OS, we set $(\hat{k}, \sigma', min) = (0.1, 300, 100)$ to get small regions as suggested by [102], Appendix A. Visual results for a single image partitioned into several regions are displayed in Table 3.9. The partitions obtained with the different methods are displayed in the first row, while the true labelled images followed by the segmentation results including their performance values $(TPR\%, OSR\%)$ appear in the second row. From results in Table 3.8, SOM was chosen as the method to establish the partition because of its performance expressed by the TPR value which becomes the highest.

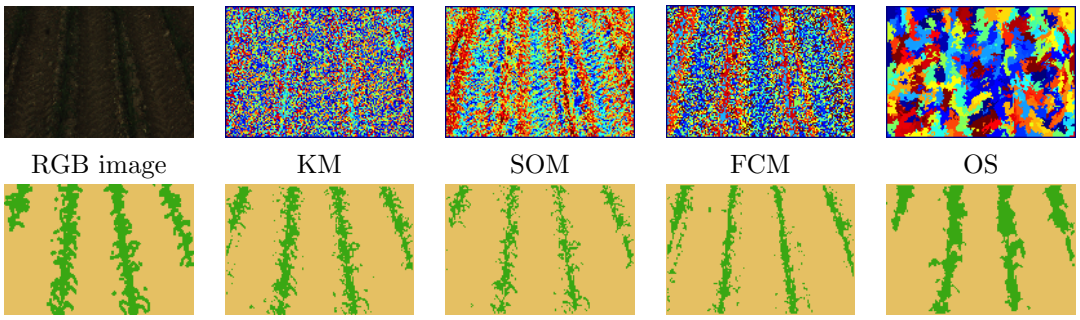


Table 3.9: First row: RGB image split into multiples regions with different algorithms. Second row: True labelled image followed by the segmentation results with their performance values $(TPR\%, OSR\%)$. Results are reported in [7].

3.4.3 Vegetation segmentation

Images in Ω_2 were processed with Algorithm 1, using SOM as the selected partitioning method and H_{CVI4} as the pattern descriptor. The proposed method was tested against different methods used in precision agriculture for image segmentation. Also, the standard method for green segmentation in Fig. 3.2a is included. The results are identified as: TH_{ExG}, TH_{ExGR}, TH_{CIVE}, TH_{VEG}, TH_{COM}, and TH_{Grey1}, the sub-index identify the transformation function used to obtain the grey image, i.e. ExG, ExGR, CIVE, VEG and COM. Additionally, two semantic segmentation methods were also included in the analysis; one based in convolutional (patch) neural networks (CNN) [114], open source library (CN24, [115]), and the second based on Random Forest Trees (RFT) [116]. The CN24 library includes a pre-trained model able to identify multiple classes in urban scenes. In our dataset, different CNN architectures were tested and the best results posted in Table 3.10, were achieved with the pre-trained model.

The segmentation quality was evaluated with the statistical metrics provided in Table 3.2. Numerical results are shown in Table 3.10. The best value was achieved with the Tewari et al. proposal [20]; 87.34% and 75.59% with OSR and TPR respectively. In the same table, the OSR values obtained with Yang et al. [24] and Hlaing and Khaing [21]

proposals are poor, below 80%. In this context, authors reported an accuracy over 95% in images where plants are well defined (usually, one plant per image). Maize images are more complex having several plants and weeds per image; this can be the reason for the low performance of these methods in our dataset.

It is well known in literature that CNNs have been tested with high performances in the segmentation tasks. In our case, we tested different architectures with the CN24 framework in the dataset used for testing. We have not found a suitable configuration able to increase its performance. RFT shows a similar performance to CN24. It is important to mention that although RFT has low performance in vegetation detection, results can be relevant in the context of crop lines detection provided that vegetation in the crop line is preserved and well limited. Finally, BoW representation achieves an OSR of 86.11% with a percentage of vegetation correctly identified of 73.24%. The rate of elements well classified is 90.39%. This value becomes enough for agricultural applications. However, the overlapping between green plants and background is 61.6% which are similar values as those obtained with the other proposals.

	TH_{ExG}	TH_{ExGR}	TH_{CIVE}	TH_{VEG}	TH_{COM}	TH_{Grey1}
OSR	86.95	85.83	83.76	85.72	75.95	87.71
TPR	71.67	83.51	74.02	67.27	53.22	74.16
TNR	89.58	85.20	85.14	88.35	90.58	89.40
PPV	66.38	40.17	44.25	60.16	73.36	63.32
NPV	93.00	97.47	95.74	92.99	78.61	94.00
F	67.10	53.05	50.27	61.29	55.59	67.67
	Yang	Hlaing et al.	Tewari et al.	RFT	CN24	**BoW**
OSR	79.40	70.64	87.34	75.29	82.64	86.11
TPR	60.25	13.65	75.59	53.90	71.43	73.24
TNR	84.08	76.60	89.84	82.04	84.89	90.39
PPV	44.28	4.23	60.31	54.16	39.71	58.60
NPV	91.08	90.44	92.49	83.95	93.72	89.51
F	44.94	3.24	64.96	52.35	43.32	61.60

Table 3.10: Vegetation segmentation performance. Metrics into rows (Table 3.2), and methods into columns. Results reported in [7].

For a single image, the vegetation segmentation obtained with methods in Table 3.10 is shown in Table 3.11. In brackets are the TPR and OSR values in terms of percentage.

In addition to the results displayed in Table 3.11, images in Table 3.12 were processed with TH_{Grey1}, Tewari et al. and BoW. They achieve the best performance in Table 3.10, with the segmented results displayed in Table 3.12.

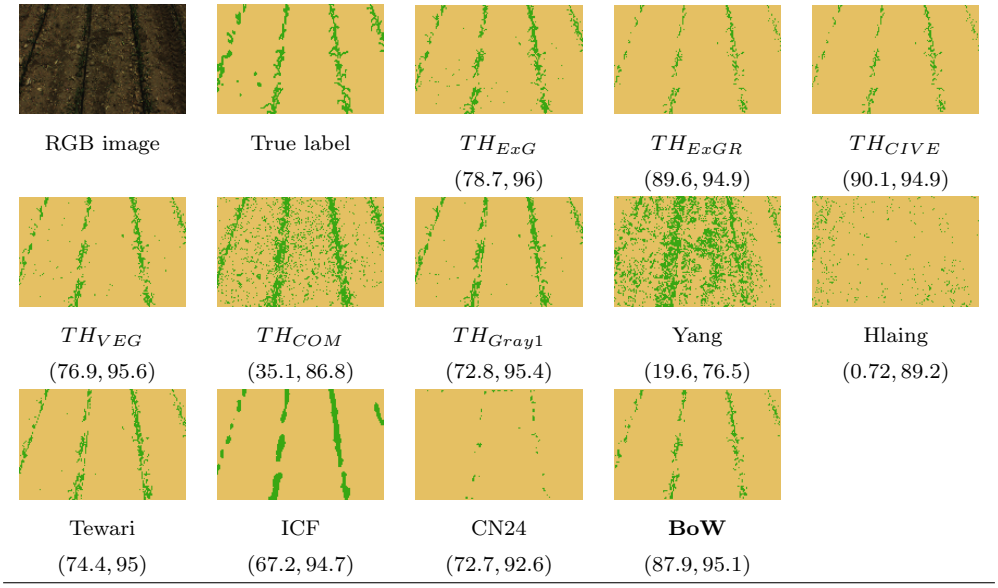


Table 3.11: Image segmentation with methodologies in Table 3.10, each result is accompanied with its $TPR\%$ and $OS\%$ values in brackets. Results reported in [7].

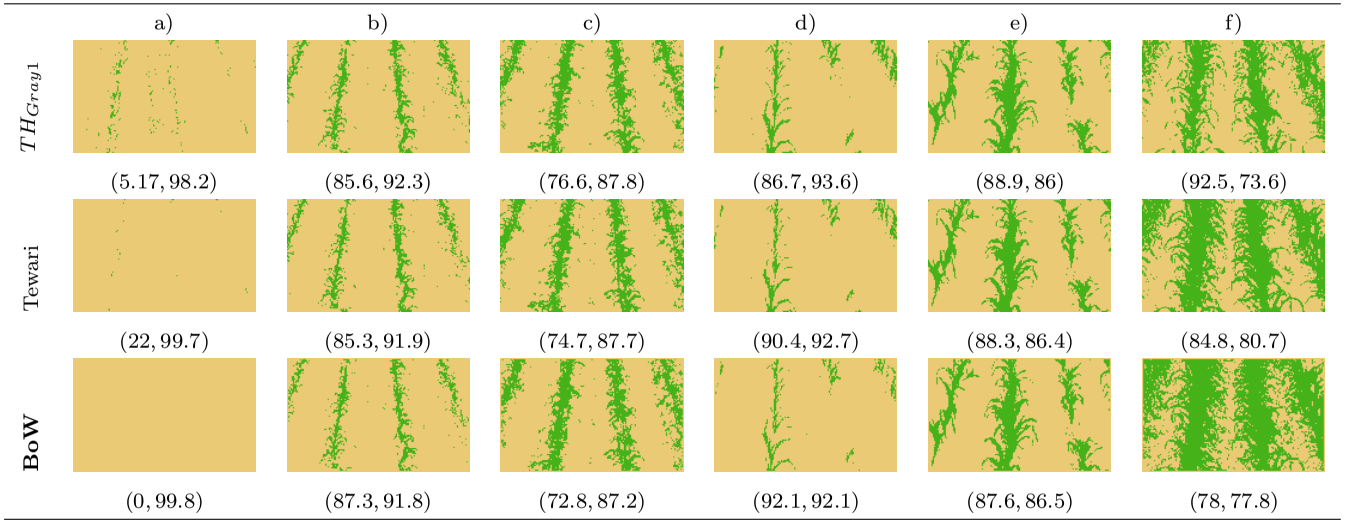


Table 3.12: Segmentation results under different scenarios. In Table 3.1, the RGB images and their corresponding true labelled image (image per column). Performance values are in brackets $(TPR\%, OSR\%)$. Results are reported in [7].

Chapter 4

Obstacle detection

In this chapter a strategy for automatic static/dynamic obstacle detection in agricultural image sequences is described. The approach consists in a spatial analysis to obtain an initial segmentation of objects of interest and subsequently, temporal information is used for discriminating between moving and static objects. To our knowledge in the field of agricultural image analysis, classical approaches make use of either spatial or temporal information, but not both at the same time, as in the proposed approach, making an important contribution in this regard. Another characteristic of the proposed method is that it does not require any training process achieving favourable results when tested under different outdoor scenarios in agricultural environments, which are really complex due to the high variability in the illumination conditions, causing undesired effects such as shadows and alternating lighted and dark areas as described in Chapter 1. The experimental strategy designed to evaluate the obstacle detection performance showed that the results are comparable outperforming those reported in current literature, Chapter 2. All results in this chapter were published in [117].

The remainder of the chapter is organized as follows: the problem statement and the main objectives are given in Section 4.1, the methodology description is provided in Section 4.2, and finally the experimental results in Section 4.3 close the chapter.

4.1 Introduction

In recent years, agricultural vehicles (tractors, combiners, sprayers, spreaders) have been introduced in agricultural environments to accomplish different agricultural tasks. In order to be useful, such vehicles should be equipped with machine vision-based sensors, which provide the required information to develop these tasks with good performances.

Additionally, vehicles need a means of detecting obstructions to avoid collisions. If the system detects people, animals, other vehicles, or any obstruction, inside the haulage area during autonomous operation, the vehicle must stop immediately for security reasons making this the main motivation of the proposed research.

The principal objective of this chapter is to propose a method for obstacle detection in crop maize to find static (trees, fence post, buildings, people standing up, etc.) and non-static (people walking, and moving tractors) elements in the autonomous tractor trajectory by processing the images acquired with the machine vision system described in Chapter 2. From now on, image sequences are referred to as video sequences due to the images can be also seen as sequences of consecutive frames. In such video sequences, all scenarios described in Table 4.1 are possible.

Obstacles state	Description
Scenario 1	The tractor is stopped and the obstacles are static
Scenario 2	The tractor is moving and the obstacles are static
Scenario 3	The tractor is stopped and the obstacles are moving
Scenario 4	The tractor is moving and the obstacles are moving

Table 4.1: Possible scenarios in the image sequences described in Chapter 2. Static obstacles: trees, buildings, people standing up, among others. Moving obstacles: people walking, moving tractors, etc..

Images from agricultural videos, containing several elements including plants, trees, weeds, soil, objects, and shadows coming from these elements, are complex from the image processing point of view. The complexity increases because of the great variability of vehicle and environmental conditions, such as: changing seasons or weather conditions (sunny or cloudy days), time of day, dust or movements produced by the vehicle in movement along the field, and also because of vibrations caused by the tractor engine. As a consequence of these combined effects, a robust obstacle detection system is very necessary.

To achieve the proposed goal, a Static and Dynamic Combined approach (SDC) based on spatio-temporal analysis has been devised. Given an input video, two consecutive video frames are considered for both spatial and temporal processes. Obstacle segmentation is made by analysing spatial information in the current frame while motion is computed from the first-order spatio-temporal derivatives of two consecutive frames, Fig. 4.1. The spatial and temporal information conveniently combined allows determining objects location and their associated movement, if any. One of the major advantages of this representation is that by analysing temporal information, not only obstacle segmentation is improved, it is also possible to identify moving obstacles in front of the tractor trajectory. Motion detection is useful to anticipate and prevent collisions between autonomous vehicles; this topic is widely discussed in [90]. When the spatial and

temporal information is combined, the complexity of motion detection is significantly reduced. Instead of detecting motion with a relatively large number of previous frames, where the scene could change dramatically, we estimate motion based on the first-order temporal difference, but compared against the obstacle segmentation results to determine which of the elements on the scene are really in movement with respect to the tractor motion.

Unlike other reported methods where object detection is exclusively based either on static segmentation or differential considerations, such as those based on optical flow computation, the SDC allows exploiting the advantages of both, making an important contribution in the context of object detection in agricultural environments. The spatial differentiation in optical flow-based methods is assumed by the temporal differentiation, which determines image changes [118]. This combination and the simultaneous separation of spatial and temporal analysis is the main contribution of the proposed approach, which is conveniently compared against existing static and dynamic strategies.

4.2 Methodology description

The aim of this work is static/dynamic obstacle detection in agricultural videos. The proposed SDC method involves a sequence of steps where spatial and temporal information is joined to achieve the goal.

The video data is processed according to the scheme in Fig. 4.1 to make a decision at each instant of time during the tractor navigation. The inputs to the system are two consecutive frames, the current frame $I(x, y, z)^t$ and the frame at the previous time instant $I(x, y, z)^{t-1}$; t is the time frame index, (x, y) the pixels coordinates and z represents a specific colour channel, according to the colour space chosen. For simplicity from now on: I^t and I^{t-1} are used to refer to the current and the previous frame.

As one can see from Fig. 4.1, the system involves two main modules where the analysis is carried out at pixel level; the first module is based on the spatial analysis where an image segmentation approach is made for obstacle detection in the current frame, I^t. The segmentation is performed in the L*a*b* colour space [2] based on foreground segmentation; then small spurious areas are removed using morphological operations. A temporal analysis is accomplished in the second module to detect the most prominent changes in the current frame applying a temporal differentiation between I^t and I^{t-1}. Spatial and temporal information are conveniently combined during the process, obtaining as a final result the static and non-static obstacles embedded in I^t. For

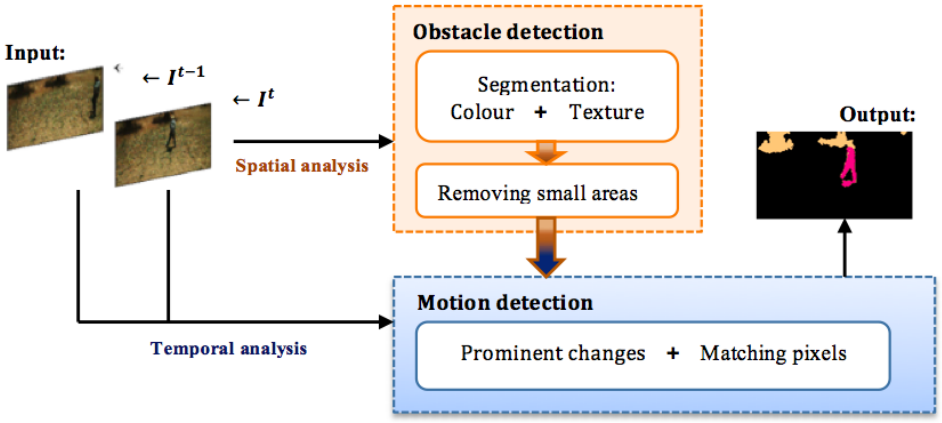


Figure 4.1: Static/dynamic obstacle detection for agricultural videos. The output contains static (brown light colour) and dynamic (pink colour) obstacles detected from input image I^t.

a better understanding of the methodology, this section is divided into two main parts. First, obstacle segmentation is explained, followed by motion detection description.

4.2.1 Spatial analysis

Agricultural outdoor images are complex to analyse, mainly because they are captured under different illumination conditions (dawn, sunset, poor or low illumination), most times with high illumination variability during the same working day because of cloud displacements causing important changes in the resulting images. The obstacle segmentation method must be robust enough to deal with these adverse situations. Under these considerations, we studied different colour spaces and found that L*a*b* space is the most appropriate to spatial analysis in agriculture images from the point of view of motion detection.

The CIELAB colour space was developed in order to obtain an easy-to-calculate colour measurement in accordance with the Munsell colour order system [2]. Furthermore, L*a*b* space is useful for comparing similar colours, which is an event included in our case study, in Fig. 4.2.

In maize field images, during the activities oriented to weed removal, the common elements are green plants (weeds and crops) and soil containing spectral components with high green and red content respectively. Because of this, the majority of obstacles in the images are detected from b* channel as it represents the mapping along blue and yellow axis, i.e. colours with low probability of occurrence in maize fields containing the natural elements (plants and soil), Fig. 4.2c. However, for this reason obstacles containing a

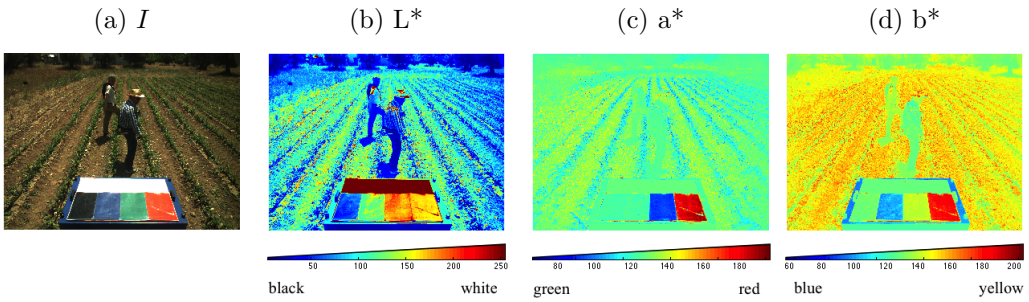


Figure 4.2: Colour image I transformed from RGB to L*a*b colour space. From b to d, the L*a*b* colour channels separately [2].

high white and/or red component are not detected from the b* channel. White and red elements need to be directly identified from L* and a* channels respectively.

Furthermore, in agricultural environments it is quite common for people to wear clothes with similar colours to the ones existing in the maize fields, i.e. with green and red-brown tones, leading to severe failures during the object detection process based exclusively on spectral information. Fig. 4.3 shows an example of this failure. The binary image Fig. 4.3-b is the result of the colour segmentation process applied over the image in Fig. 4.3-a. As can be seen, the person on the left of the image is not detected due to their clothes have similar colours to the soil; this problem was also reported in [84]. This is because texture information (Fig. 4.3-c) is required in order to improve object detection as shown in Fig. 4.3-d.

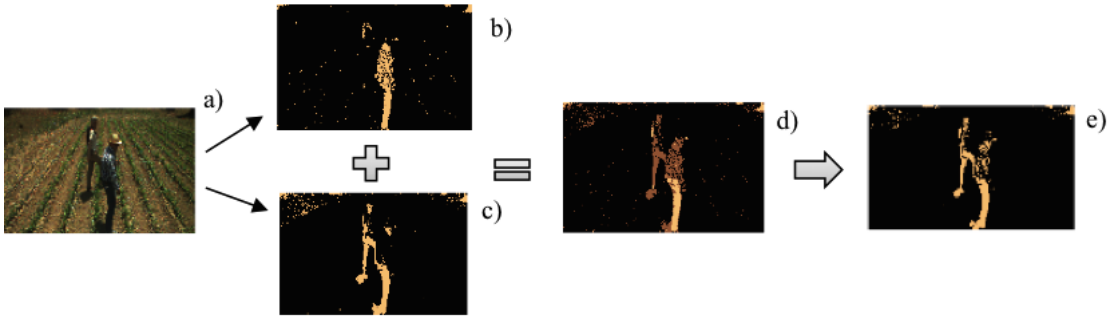


Figure 4.3: a) Colour image. b) Segmentation based on L*a*b* colour space. c) Texture extraction. d) Segmentation combining b and c. d) Final result after removing small areas.

The full scheme of the proposed obstacle segmentation algorithm is portrayed in Fig 4.4, where a colour image is split into two complementary sets of pixels: the background, containing pixels from green plants and soil, and the foreground, which contains the remaining pixels. From top to bottom, white (I_{white}), red (I_{red}) and foreground (I_{obs}) detected pixels from L*, a* and b* channels respectively, followed by the texturized image. From which pixels having different texture to soil and green plants are extracted (I_{btex}). The information coming from I_{white}, I_{red}, I_{obs}, and I_{btex} is concatenated into a

single image. The resulting binary image contains small and spurious areas; they are removed to retain the main areas, which are potential objects. The final binary image I_{bin} can be appreciated at the bottom in Fig 4.4. The complete process to obtain these intermediate images is outlined in the steps 1a, 2a and 3a, bellow.

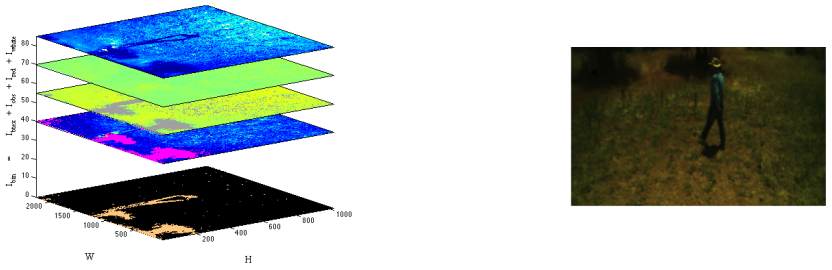


Figure 4.4: Obstacle detection process applied to the colour image (H-by-W size) on the right.

Before proceeding, it is necessary to define the statistic measures used to compute the threshold values τ_1 and τ_2 for segmenting an image of size $H \times W$. Let $I(x, y)$ be the intensity of pixel (x, y). The mathematical definition of the mean (μ), standard deviation (σ) and skewness (s) are given by expressions 4.1, 4.2 and 4.3 respectively. The threshold values τ_1 and τ_2 are calculated by way of Equations 4.4 and 4.5, where n is the number of standard deviations considered.

$$\mu = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} I(i,j) \tag{4.1}$$

$$\sigma = \sqrt{\frac{1}{H \times W - 1} \sum_{i=1}^{H} \sum_{j=1}^{W} (I(i,j) - \mu)^2} \tag{4.2}$$

$$s = \frac{\frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} (I(i,j) - \mu)^3}{\left[\frac{1}{H \times W - 1} \sum_{i=1}^{H} \sum_{j=1}^{W} (I(i,j) - \mu)^2\right]^{\frac{3}{2}}} \tag{4.3}$$

$$\tau^1 = \mu + n \times \sigma \tag{4.4}$$

$$\tau^2 = \mu - n \times \sigma \tag{4.5}$$

Continuing with the methodology description. Given the current frame I^t in RGB colour space follow Steps 1a, 2a and 3a to achieve the obstacle segmentation desired.

Step 1a. Spectral segmentation

First, transform RGB image into L*a*b* colour space to obtain L*(I_L), a*(I_a) and b* (I_b) channels respectively.

Apply Algorithm 2 over I_b to get a binary image I_{obs} where, pixels with value "1" represent obstacles and pixels with value "0" represent background. As mentioned before, b* channel is related to the blue-yellow axis and for this reason Algorithm 2 does not capture objects with high red spectral component values. To emphasize objects composed of red pixels we make use of I_a. In a*-channel, green colour is in the negative axis while red colour is in the positive axis. Red pixels are filtered with the expression: $I_{red}(x,y) = 1 \quad \text{if} \quad I_a(x,y) > \tau_a^1, \quad 0 \quad \text{otherwise}$. Note that the threshold value τ_a^1 is computed from the intensity values in I_a using Eq. 4.4. In order to capture white elements, a similar process with red detection is made. From luminosity channel next operation is applied: $I_{white}(x,y) = 1 \quad \text{if} \quad I_L(x,y) > \tau_L^1, \quad 0 \quad \text{otherwise}$.

Algorithm 2: Process to extract the foreground in agricultural images.

Input : I_b, b*- channel from L*a*b* colour space.
Output: I_{obs}, 1: foreground/obstacles, 0: background

1 From I_b compute; the frequency histogram (h_b) [119], the skewness s_b and the threshold τ_b^1 values (Eq. 4.3 and 4.4) ;
2 Apply finite differences method [120] to compute the critical points from h_b ;
3 Calculate the threshold value, $thl =$ Global maximum critical point ;
4 **if** $s_b \leq 0$ **then**
5 | **return** $T = thl$;
6 **else**
7 | $thl_1 =$ Minimal critical point before the global maximum critical point;
8 | return $T = min(\tau_b^1, thl, thl_1)$;
9 Threshold image as: $I_{obs}(x,y) = \begin{cases} 1 & \text{if } I_b(x,y) < T \\ 0 & \text{otherwise} \end{cases}$;
10 **return**;

Step 2a. Texture extraction

The texture content in the image $I(x,y,z)^t$ is computed from the greyscale image $I_{GREY}(x,y)$. For each intensity value in I_{GREY}; the difference between the maximum and minimum grey level over a $\omega - by - \omega$ neighborhood $\Omega(x,y)$ centred on the interest pixel (x,y) defines the texture value of the pixel, Eq. 4.6. From the texture image I_{tex}, the segmentation is carried out as: $I_{btex}(x,y) = 1 \quad \text{if} \quad I_{tex}(x,y) < \tau_{tex}^2, \quad 0 \quad \text{otherwise}$. The threshold value τ_{tex}^2 is computed with Eq. 4.5.

$$I_{tex}(x,y) = max\left(I_{GREY}(x_i,y_i)\right) - min\left(I_{GREY}(x_i,y_i)\right), (x_i,y_i) \in \Omega(x,y) \tag{4.6}$$

Information coming from spectral segmentation and texture extraction is combined to achieve the full obstacle's segmentation I^*_{bin}, Eq. 4.7.

$$I^*_{bin}(x,y) = (I_{red}(x,y) + I_{white}(x,y) + I_{obs}(x,y) + I_{btex}(x,y)) > 0 \tag{4.7}$$

Step 3a. Small areas removal

Once I^*_{bin} is obtained containing potential obstacles, small or spurious regions are suppressed under the assumption that they are irrelevant from the point of view of safe autonomous navigation. To perform this task a morphological opening [121] is carried out in I^*_{bin} obtaining as a result the binary image I_{bin}. It contains the relevant obstacles on the current frame $I(x,y,z)^t$. Opening is the combination of two fundamental morphological operations, an erosion followed by a dilation. This allows removing small spurious binary areas, isolated pixels or pixels on object boundaries. The number of pixels removed from the objects in an image depends on the size and shape of the structuring element used to process the image. In this research, the structuring element chosen for both operations was a circle with different sizes, as discussed below.

Due to the camera imaging perspective projection, Fig. 2.1 Chapter 2, the same object in a frame located at different spatial positions in the 3D scene is mapped onto the image with different sizes, measured in pixels; its size is small at the top and big at the bottom of the image. For this reason, erosion is not uniformly carried out on the binary image. Circles with different radio lengths were used in the erosion, taking the image height as the reference. Four areas are identified, from the top to the bottom; the erosion was performed using circles of radio 1, 2, 3 and 4 pixels respectively, which are sufficient according to the camera vision system setup [95]. Dilation was uniformly performed throughout the whole image using a circle of radio 1. The result after removing small areas in the segmented image in Fig. 4.3-d is shown in Fig. 4.3-e, as an illustrative example.

The next phase consists in identifying which elements on I_{bin} are stationary or moving. For this task temporal information is required to identify static and non-static pixels associated to such elements.

4.2.2 Temporal analysis

There are many existing schemes in literature for motion detection; they were described in Section 2.1.3. The simplest and most popular approach involves two consecutive frames. This is the basis of the proposed approach for discriminating among static and non-static obstacles in maize field videos. We consider the information provided by the previous frame I^{t-1} to detect moving pixels in I^t using as reference the obstacle segmentation mask I_{bin}. Now, discrimination among static and non-static obstacles is described according to the following steps.

Step 1b. Prominent motion detection

The temporal gradient is extracted using the first order differential approximation: $G^t = I^t - I^{t-1}$. The prominent changes in G^t are obtained by thresholding. From G^t, two threshold values, τ_G^1 and τ_G^2, are computed with the expressions 4.4 and 4.5 respectively on the image G^t. These values are used to get two binary images d_1^t and d_2^t by Eq. 4.8 and Eq. 4.9 respectively at pixel location (x, y).

$$d(x,y)_1^t = \begin{cases} 1 & \text{if } G^t(x,y) > \tau_G^1 \\ 0 & \text{otherwise} \end{cases} \tag{4.8}$$

$$d(x,y)_2^t = \begin{cases} 1 & \text{if } G^t(x,y) < \tau_G^2 \\ 0 & \text{otherwise} \end{cases} \tag{4.9}$$

In Fig. 4.5, the temporal gradient from two consecutive frames is portrayed, their respective resulting binary images d_1^t and d_2^t are displayed in Fig. 4.6. As shown in the images, pixels with major changes at the scene can be detected.

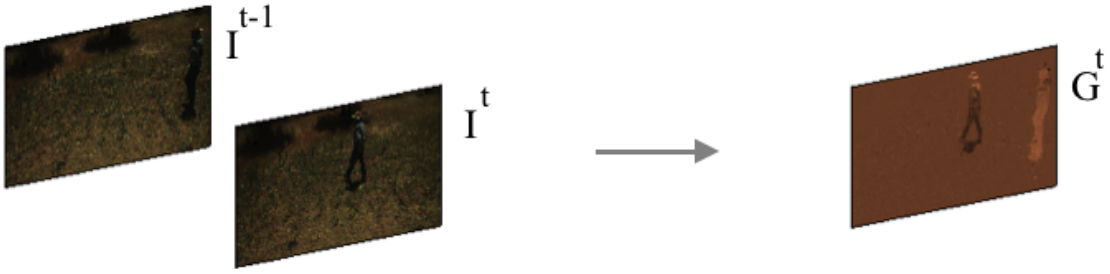


Figure 4.5: Left: two consecutive frames; I^t and I^{t-1}. Right: the temporal gradient G^t computed from I^t and I^{t-1}.

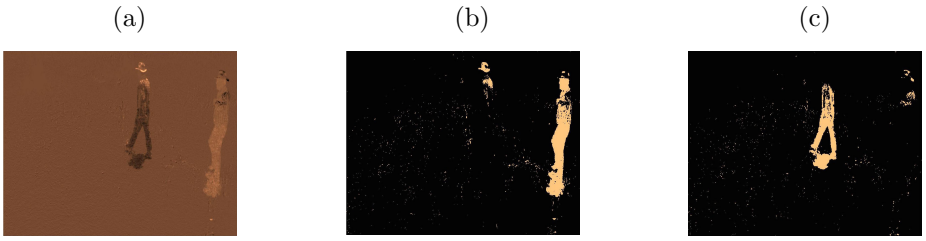


Figure 4.6: a) Temporal gradientG^t, Fig. 4.5; b, c) Binary images d_1^t and d_2^t.

Step 2b. Matching process

To determine which of the two masks is the most convenient, i.e. the one containing the maximum movement information in the current frame, the following is performed: d_1^t and d_2^t are independently overlaid on top of the obstacle in the segmented image I_{bin}. From each overlapping the number of matching pixels is counted using expressions 4.10 and 4.11 giving as results M_1 and M_2. The mask with a major number of matching pixels $max(M_1, M_2)$ is chosen as the moving mask for the current frame I^t.

$$M_1(I_{bin}, d_1^t) = \sum_{i=1}^{H}\sum_{j=1}^{W}\{d(i,j)_1^t \mid d(i,j)_1^t + I_{bin}(i,j) > 1\} \tag{4.10}$$

$$M_2(I_{bin}, d_2^t) = \sum_{i=1}^{H}\sum_{j=1}^{W}\{d(i,j)_2^t \mid d(i,j)_2^t + I_{bin}(i,j) > 1\} \tag{4.11}$$

Continuing with the example in Fig. 4.6, d_2^t contains the greatest number of matched pixels with I_{bin}; $M_1 < M_2$. This can be seen in Fig. 4.7, I_{bin} is on the base of the two graphics, with d_1^t (on the left) and d_2^t (on the right). Pixel pairs matching are highlighted in pink colour at the base (bottom) and on top of the graphics. To conclude, also masks on the base of the graphics show static (light brown colour) and non-static (pink colour) pixels.

4.3 Experimental results

A set of 110 videos of different length, the longest with 2099 frames while the shortest with 20 frames, with a total of about 15000 frames, compose the dataset including videos during the final demonstration of the RHEA project in May 2014. There are four types of events that can be observed in the dataset, they are described in Table 4.1.

It is worth noting that in scenarios 1 and 3, although the tractor is stopped, small movements are detected, derived from the motor vibrations when it is turned on. In

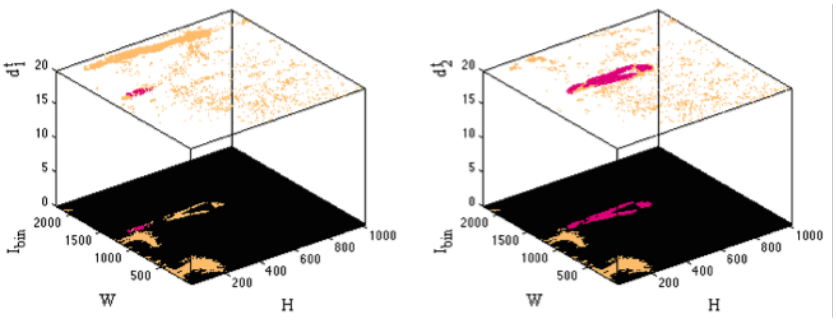


Figure 4.7: Matching pixels for motion detection: Left: I_{bin} on the base, on top in light brown colour and transparent background d_1^t (Fig. 4.6b). Right: I_{bin} on the base and d_2^t (Fig. 4.6c) at the top. Pixel pairs matching are highlighted in pink colour.

order to provide a quantitative evaluation of the proposed method, statistical measures described in Table 4.2 are used to determine the performance of the strategy. The terms; TP, FN and FP are the true positive, false negative and false positive outcomes.

Name	Abbreviation	Equation
Sensitivity or true positive rate	TPR	$TP/(TP+FN)$
Precision or positive predictive value	PPV	$TP/(TP+FP)$
False negative rate	FNR	$FN/(FN+TP)$
False discovery rate	FDR	$1-PPV$

Table 4.2: Performance evaluation metrics.

There are two parameters involved in the whole process; the number of standard deviations considered in Eq. 4.4 and 4.5, and the window size related to the texture extraction, Eq. 4.6. In the first case, n is set to 2. Experimental results have shown a good efficiency with this setting for all experiments carried out. Regarding the window size ω required to define the neighbourhood area for texture extraction, this value must be conveniently fixed. If the block size is too large, the probability that the region contains different textures is high. On the other hand, if the block size is too small, the capture of sufficient texture information becomes difficult. It is not easy to define an optimum block size. We tested different sizes with $\omega = \{3, 5, 7, 9, 11\}$ where a trade-off between the image resolution and the block size must be established. After different trial and error experiments we have verified that with the camera system setup and focal length the best performance was achieved with $\omega = 9$.

From now on, quantitative and qualitative results are given. Firstly, obstacle segmentation outcomes are provided. Secondly, motion detection as well as the evaluation of the discrimination among static and non-static obstacles under all scenarios described in Table 4.1 is presented. In both cases, performance is compared against well-known methodologies in the state-of-the-art provided in Section 2.1.3. To close,

results were computed using the Image Processing Toolbox MATLAB 2016b for 64 bits under Windows 10 and Intel Core 2 CPU, 3 GHz, 4 GB RAM.

4.3.1 Obstacle segmentation

To measure segmentation quality, the Success Rate (SR) is obtained as follows. Given a ground Truth Bounding Box (TB) and the segmentation result (S) achieved with the process described in Section 4.2.1, SR is computed as area $area(TB \bigcap S)/area(TB)$. The process is illustrated in Fig. 4.8 using a single image where two obstacles are present In the frame. In Fig. 4.8b the ground truth-bounding box for the obstacles are shown in Fig. 4.8a while in Fig. 4.8c the segmented image is obtained with the proposed approach. The SR achieved for obstacles 1 and 2 were 97.65% and 97.55% respectively.

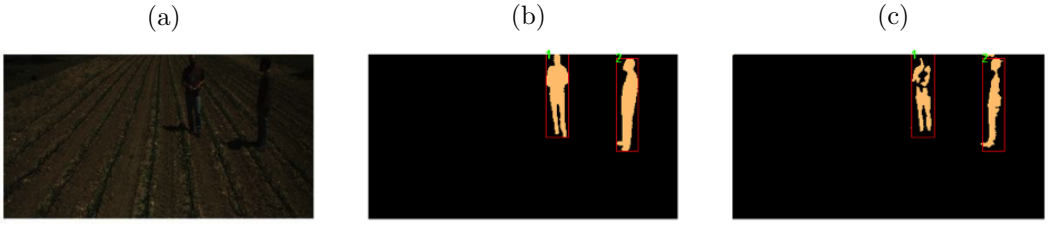


Figure 4.8: a) Colour image. b) Obstacles on the left image manually segmented. They are enclosed in red boxes and represent the ground-truth (TB). c) Obstacles detected with our segmentation algorithm (S); the SR values are 97.65% and 97.55% for obstacles 1 and 2 respectively.

Now, with respect to the evaluation for each case in Table 4.1, the following process is carried out. From the video dataset available, 80 frames were randomly chosen. This selection includes 20 images per scenario. The regions considered as obstacles were manually touched up, pixel by pixel, to delimit and fill gaps at each obstacle. A bounding box is drawn for each obstacle representing the TB. The total number of true bounding boxes obtained was: 31, 36, 29 and 51 for scenarios 1, 2, 3, and 4, respectively (second column of Table 4.3). After that, the 80 images were processed by applying the algorithm described in Section 4.2.1 to obtain the corresponding obstacle segmentation (S). Results were compared with the truth bounding boxes to get the SR for each obstacle. Finally, the average success rate for each scenario is presented in the third column of Table 4.3. Similar to the process previously described, we obtained the average SR results using Otsu [122] and Fuzzy C-Means [123], segmentation algorithms. Results are displayed in the fourth and fifth columns in the same table.

Now, for a broader evaluation of how the obstacle segmentation process is working, the number of images tested was increased. The output of a set of 300 frames was visually inspected to determine when a success occurs ($TP-$obstacle detected) or when a failure takes place ($FN-$obstacle non-detected, $FP-$ non-existent obstacle detection).

Case	# Obstacles	Our segmentation approach	Otsu	FCM
Scenario 1	31	74.96	16.39	66.14
Scenario 2	36	85.93	20.31	66.12
Scenario 3	29	85.61	24.39	65.72
Scenario 4	51	86.38	19.70	63.14

Table 4.3: Average success rates (%) achieved with different segmentation processes.

The numerical evaluation results from the expressions in Table 4.2 are shown in Table 4.4. The obstacle detection with the process described in Section 4.2.1 shows good performance. The proportion of obstacles correctly identified is 93.88% while the percentage of positive predictive values is 83.36%. Accuracy rate is similar to the results reported by Reina and Milella [77].

Metric	Evaluation (%)
TPR	93.88
PPV	83.36
FNR	6.12
FDR	16.64

Table 4.4: Evaluation of the metrics in Table 4.2 for the obstacle detection algorithm, Section 4.2.1.

Additional obstacle detection results are provided in the Fig 4.9, four consecutive frames from a video sequence are displayed in the first row. Trees detected at each frame using the proposed approach are in the second row. Now we have outlined the contours defining the objects instead of the binary results. Stems can be used as references for path planning of the autonomous vehicle, as suggested in [70, 90], and [89] where stems are detected as part of the navigation system.

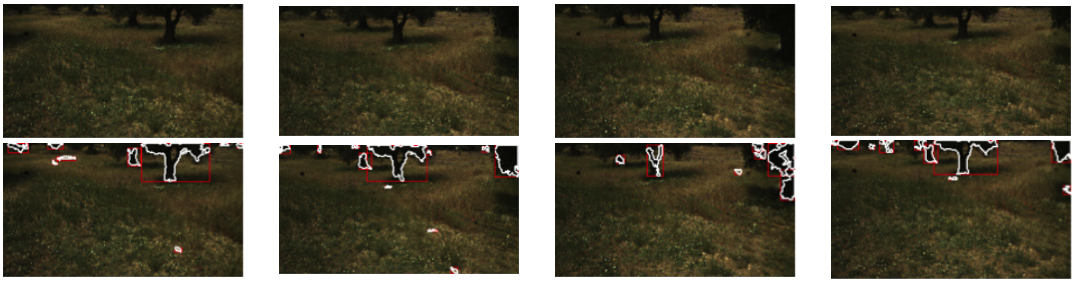


Figure 4.9: Consecutive frames. Trees trunks detected are delimited by using white lines and enclosed in red boxes.

Up to this point the results related to object obstacles detection based on spatial analysis have been given. Motion detection results are thoroughly discussed below.

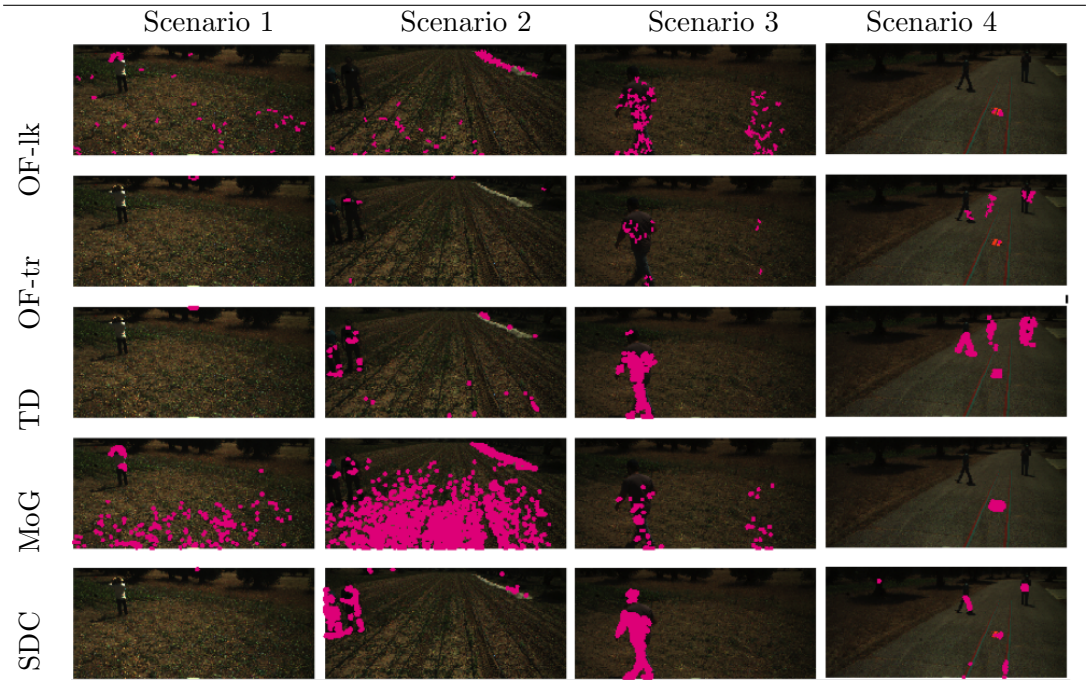
4.3.2 Motion detection

The proposed approach is compared against four well-known methodologies reported in literature: Optical Flow (OF) estimation based on Lucas–Kanade (Of-lk) [124], Tikhonov regularization (Of-tr) [124], Temporal Differencing (TD) [70], and Mixture of Gaussians (MoG) [71]. Details concerning these four methods can be found in the referenced papers. For optical flow results, the pixels with major optical flow values were filtered by a threshold value of 0.5 as suggested in [124].

All methods including the proposed method were tested with consecutive frames in the four scenarios considered above. Some visual results are displayed in Table 4.5 with illustrative purposes. In Table 4.5 methods appear in the rows while the scenarios are displayed in the columns. The pink colour labels in the images represent motion detection. The ideal outcome in this table should be when no pink marks appear in scenarios 1 and 2; in contrast, to scenarios 3 and 4, pink colour must only be applied on the moving obstacles. The best outcome in Table 4.5 occurs when the tractor is stopped, scenarios 1 and 3. In these scenarios, the performance of OF-tr, TD and SDC is acceptable, from the point of view of agricultural applications, while OF-lk and MoG fail because motion is detected on static regions. The poor results, including the proposed methodology, occur when the tractor is operating, i.e. moving. This failure is clearly noted in scenario 2 where motion is wrongly detected on regions where nothing happens. Here, the best outcome is achieved with OF-tr. The weak point of this approach occurs on non-static obstacles as can be seen in scenario 4 (second row, fourth column). The method OF-tr has problems to remove motion that belongs to the previous frame. In this regard, SDC works well because the motion mask is selected using the obstacle segmentation in the current frame as reference. This information is applied for motion removal coming from the previous frame. In the opposite case, the proposed approach still achieves a poor level of performance in scenario 2. The consecutive frames of examples in Table 4.5 can be seen at the first and second rows in Table 4.7.

The qualitative results in Table 4.6 reinforce the foregoing. In this table, the qualitative performance evaluation of motion detection is obtained by counting the number of successes achieved in the process to identify static and non-static obstacles in the agricultural video sequences. For scenarios 1 and 2, an event is considered a success if no motion is detected. In contrast, if motion is detected then the event is considered a failure. In scenarios 3 and 4, the event is a success in any of these cases: motion is detected on static elements or motion is detected on moving obstacles. A fault occurs when motion is detected in static elements or motion is not detected in moving obstacles. The quantitative performance is expressed in terms of percentage of success. From Table

Table 4.5: Motion detection in consecutive frames for scenarios in Table 4.1. The consecutive frames can be seen in the first and second rows in Table 4.7.



4.6 also OF-tr and SDC achieve the best average successes in terms of percentage, the first for scenario 2 and the second in the remaining scenarios. In this table it is important to highlight that the accuracy in scenario 4 using OF-tr decreases significantly because the number of non-static obstacles is larger than static obstacles in videos for scenario 4.

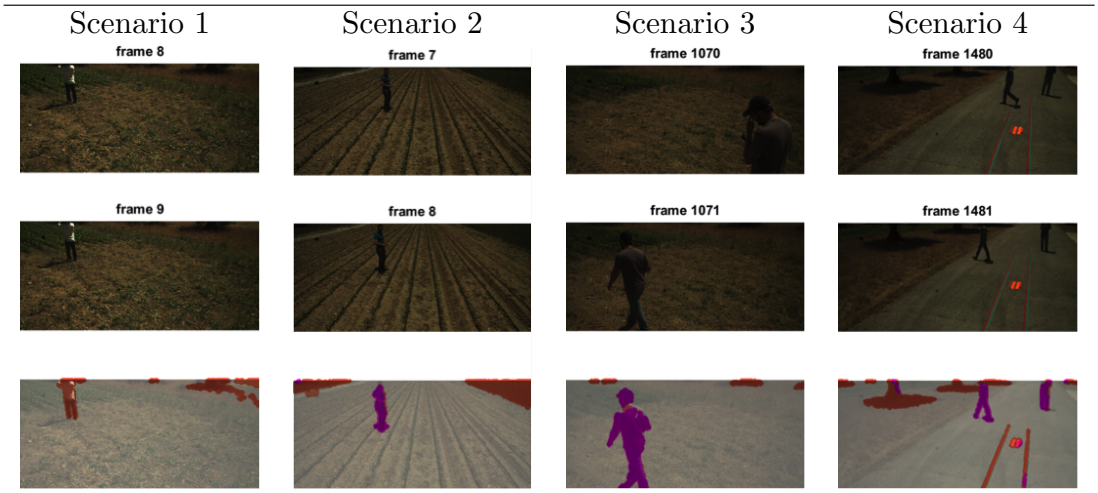
Table 4.6: Percentage of success for motion detection based on the methodologies in Table 4.5.

	OF-lk	OF-tr	TD	MoG	SDC
Scenario 1	55.14	79.83	88.88	54.15	98.78
Scenario 2	66.08	86.52	54.05	22.88	61.11
Scenario 3	61.12	78.68	76.54	30.22	89.88
Scenario 4	59.68	32.41	28.77	39.92	75.20
Average success percentage	60.51	69.36	62.06	36.79	81.24

It should be noted that in the fifth row in Table 4.5, results with our SDC method, only motion detection is displayed. Results, including the obstacle segmentation, are provided in Table 4.7, where obstacle segmentation is labelled with light brown colour and motion detection is highlighted with pink colour. In this table, detected motion is the same as the last row in Table 4.5. Also in this table, the two consecutive frames used to get results in Table 4.5 are given in the first and second rows.

The results hereafter refer to the performance assessments of the discrimination between static and non-static elements; metrics in Table 4.2 are also used for this task. Definitions

Table 4.7: First and second row: consecutive frames under the different scenarios. Last row: obstacle detection (in light brown) and motion detection (pink colour) obtained with SDC algorithm.



for TP, FN and FP, are different for static and dynamic obstacles and are given in Table 4.8.

True state	Apply	Terminology	Definition
Static obstacle	Scenarios 1 and 2	True positive (TP)	No detected motion in the obstacles, they are correctly static
		False positive (FP)	No detected static obstacles on the frame
		False negative (FN)	Detected motion in the obstacles given that obstacles are static
Dynamic obstacle	Scenarios 3 and 4	True positive (TP)	Detected motion in the obstacles given that they are moving
		False positive (FP)	No detected moving obstacles on the frame
		False negative (FN)	No detected motion in the obstacles given that they are moving

Table 4.8: Definitions for metrics in Table 4.2. They are used to evaluate the performance of our SDC algorithm to discriminate among static and non-static elements.

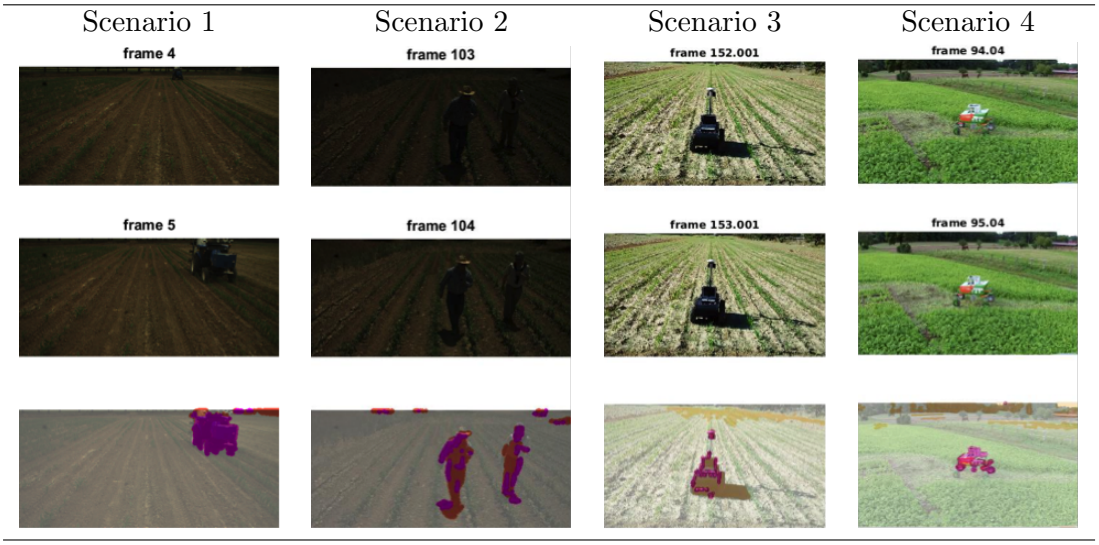
After applying the proposed SDC approach to the videos database, results were visually inspected to determine when the method performs successfully or fails. The results are reported in Table 4.9. As earlier noted, the method shows good performance to detect between static and non-static obstacles when the tractor is a static state (scenario 1 and 3). However, performance is poor when the tractor navigates (scenarios 2 and 4). Still, it is difficult to identify if the motion flow comes from the tractor motion or from dynamic obstacles.

Metric(%)	Scenario 1	Scenario 2	Scenario 3	Scenario 4
TPR	93.83	34.29	91.89	63.03
PPV	96.75	39.85	82.88	80.33
FNR	6.16	19.32	8.10	36.97
FDR	3.25	13.76	17.11	19.67

Table 4.9: Average performance evaluation of the SDC algorithm to discriminate among static and non-static obstacles.

Trying to improve the comparative analysis we tested the proposed method on videos from others projects [8, 9], results are provided in Table 4.10. Images displayed in column 3 were obtained and used in [8] for path planning. Images in the fourth column were acquired from [9]. The video shows manoeuvres and applications for research robot platform "BoniRob" [78]. These images are courtesy of authors in these references.

Table 4.10: Different obstacles detected with SDC, our detector successfully detects people and obstacles in different sample fields. Images in the third and fourth columns were respectively acquired from [8] (courtesy of A. Ribeiro and co-authors) and from [9] (courtesy of Arno Ruckelshausen and co-authors).



Images in Fig. 4.10 come from a stereo-vision device. They were tested and included in the experimental results section in [3]. The SDC output is visually similar to those exposed in [3]. Our results are also comparable to those presented by Ross et al. [83] where the obstacle was a person. As can be seen, the proposed approach is capable of detecting obstacles in different crop fields and under different lighting conditions. The combination of temporal and spatial information has proven sufficient for obstacle detection, mainly for people appearing in the crop field.

Another important consideration is that the proposed approach can handle favourably the challenge of working under high variability of lighting conditions. Nevertheless, shadows are still a problem, causing failures during object detection. Table 4.11 shows

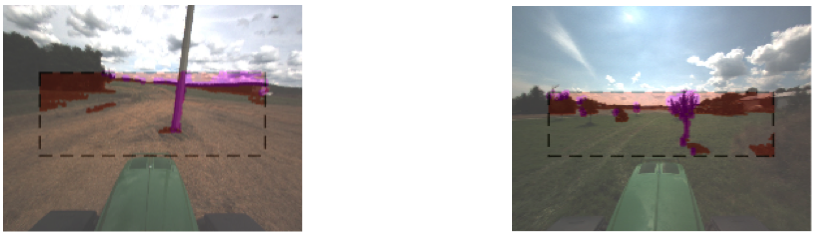
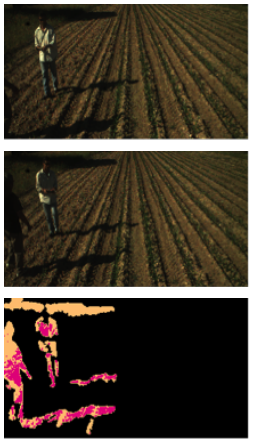
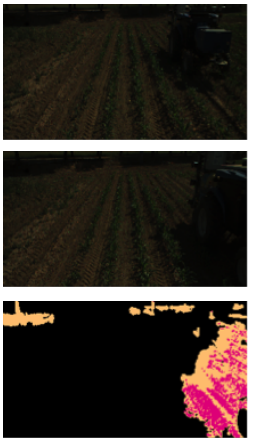


Figure 4.10: Obstacle detection in pair of images acquired by a stereo-vision device. SDC process was applied to the region enclosed by dotted black lines. Obstacles detected are in brown, while motion detection is in pink. Images courtesy of P. Fleischmann [3].

some examples where the proposed approach fails, shadows are detected as obstacles. The behaviour explained with these illustrative examples has also been seen for the set of images analysed. Wrong detection of shadows as obstacles was included in the statistical results provided in Table 4.9.

Table 4.11: Examples of shadows detected as obstacles using SDC algorithm.

Scenario 2	Scenario 2	Scenario 3	Scenario 4
			

Chapter 5

Conclusions and future work

Two relevant topics in AP have been studied in this research: vegetation and obstacle detection. These are still open issues demanding special attention in view of the commercialization of AAV in the future. As a result of this work, two algorithms were proposed providing alternative solutions to these problems. The inputs of these methods were agricultural images captured with cameras mounted on board tractors. To close this work, conclusions and future work are presented in the following paragraphs.

Vegetation segmentation

Green identification is an important step towards developing different activities in PA, where the performance in vegetation segmentation reported in current literature was over 90%. In general, the methodologies are based on thresholding techniques where the threshold value is selected from the spectral signatures (colour) properties of the images, Yang et al. [24], 95%; Guijarro et al. [25], 92%; Ye et al. [62], 92.29%; Moorthy et al. [26], 87%; Montalvo et al. [27] 91.9%. In most of these works, the evaluation and performance of the methodologies were based on ground-truth images manually labelled by using image-editing software to identify soil, crop and weed pixels. An alternative method based on BoW representation has been proposed in Chapter 3 achieving a performance in terms of classification rate over 95%, similar to those values reported in current literature. However, the proposed segmentation method needs additional improvements. This is because although the classifier achieved good performance, the segmentation designed algorithm depends on the method used to get the ROIs.

Now, regarding feature selection, a machine-learning strategy was designed where a SVM classifier model was used, which has proved able to discriminate between three classes of elements in the agricultural images, i.e. plants, soil and objects. Analysis included

assessment and study of different texture/colour/SIFT-SURF/BoW descriptors finding that the BoW representation is the most appropriate approach to characterize vegetation in the dataset used. One drawback of the learning-based strategy is that it needs manual interaction for obtaining training data. Manual segmentation is subjective depending on the supervisor performing the task, due to the lack of a standardized segmentation process in real scenarios. In this regard, the segmented training data is incorrectly or at least not perfectly labelled. In this context, the dataset generated with their corresponding handmade labelled images is open access and available online with the aim that future researchers can use it as reference for analysis. This set consists of 168 images and their corresponding handmade-labelled images [96]. They are part of the contributions of the work described in Chapter 3.

The proposed approach can be easily extended and applied to different crop fields with wide (e.g. tomato, garlic) and narrow (cereal) crop rows requiring minimal adjustment. Perhaps in cereals, where the soil may be fully covered by plants (crop and weeds), only two classes should be considered. In order to improve the performance, the following are potential improvements that can be applied:

a) Different classification techniques can be considered for classification, the SVM was used because of its good reported performance in agricultural contexts and other environments, but also different alternatives could be still feasible (e.g. K-Nearest Neighbours, Self-Organizing Maps, convolutional neural network).

b) The use of information coming from other sensors, other than machine vision, could increase the efficiency of Algorithm 1, fusing the information. It is commonly accepted that using different sources of information the performance increases. A common use of information is the one provided by near-infrared sensors to deal with CVIs involving this spectral information.

c) The use of probabilistic models [125] to improve the image segmentation results represents another alternative.

d) A thorough analysis of results obtained with the random forest trees method [116]; where the segmented images are promising for crop line detection.

Obstacle detection

The method proposed in Chapter 4 for obstacle detection in agricultural videos is based on image processing analysis supported by the combination of spectral segmentation and temporal differences. Firstly, obstacle detection is performed in the spatial scale

and then, a temporal differencing is carried out for motion detection. This approach addresses the problem of uncertainty in uncontrolled outdoor and dynamical agricultural scenarios involving autonomous machinery. Although the proposed approach could be considered deterministic in nature, the high variability in the illumination conditions, camera vibrations, unknown and unexpected static or dynamic objects introduce some uncertainty. Hence, the proposed computer vision-based approach deals well with adverse environmental conditions and specific and intrinsic situations.

The experimental results show that the obstacle detection method is suitable and effective in the context of our agricultural scenarios. We also show that trees detection is possible from the point of view of image analysing processing, making it a useful contribution for path following and mapping tasks in forestry environments. Additionally, another important contribution is that no training process is required. The method is entirely based on a combination of digital and image analysis techniques.

As future work, new research lines have been identified with high probability of improving the performance hereby presented:

a) The use of OF techniques combined with the proposed approach may be beneficial to discriminate between static and non-static obstacles. This is supported by results in Table 4.5 and Table 4.6.

b) The segmentation algorithm described in Section 4.2.1 has been improved with temporal analysis. However, the results also show that the method is not able to handle shadows, even people shadows. The process still fails on this context; shadows are difficult to identify, requiring an additional effort in this regard. The method proposed by Lalonde et al. [126] for shadow removal in outdoor images could be an alternative.

c) Animals, people, cars, fence posts, trees, buildings, and so on, need to be correctly classified in order to gain manoeuvrability and autonomy for vehicles in the field. A strategy similar to the one used for vegetation segmentation may be applied to build a model for objects discrimination by using SVM, RFT or CNN.

d) Motion detection can be used for performing a tracking task of dynamic objects. More specifically, prediction of activities to detect people walking, standing or running in the crop fields can be included to increase safe navigation.

e) As before, the use of additional information to be conveniently fused is always suitable. In this regard, in the context of object detection, the use of range-based technologies, such as LIDAR could be an interesting option as in the RHEA project.

Convincing solutions to all of these problems are required to reduce the uncertainty introduced by all factors affecting the machine vision approach, including the ones proposed for future research.

Appendix A

Partitioning methods

A short description of the techniques used for partitioning a colour image I of size $H \times W$ into p regions/clusters is provided here. Nevertheless for a more in-depth study the associated references are a good source.

For simplicity, the vectorization of the colour image I is used to explain the partitioning methods: $X = \{x_1, x_2, \ldots, x_N\}$, $i = 1, 2, \ldots N = H \times W$, where each item contains the spectral colour value of each pixel $x_i = \{r_i, g_i, b_i\}$.

K-means

Initially introduced by MacQueen in 1967 [99], it is one of the most popular methods in cluster analysis. The iterative process, Algorithm 3, allows finding the centre positions of the clusters $\bar{C} = \{\bar{c}_1, \ldots, \bar{c}_p\}$ that minimize the within-cluster sum of squares according to expression A.1. In Chapter 3, Table 3.9 second column, can be seen an illustrative example of an agricultural image partitioned with K-means.

$$\min_{\{\bar{c}_1, \bar{c}_2, \ldots, \bar{c}_p\}} \sum_{i=1}^{p} \sum_{j=1}^{|c_i|} \left(\| \, x - \bar{c}_i \, \| \right)^2 , x \in c_i \tag{A.1}$$

Despite its iterative nature, which requires convergence in its training phase, the method is fast enough from the point of view of computational cost and must be considered that different initial partition can result in different final clusters. A simple graphical representation is shown in Fig. A.1.

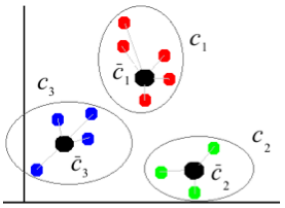


Figure A.1: Graphical representation of K-means method to split $N = 12$ items into three clusters $C = \{c_1, c_2, c_3\}$. $\bar{C} = \{\bar{c}_1, \bar{c}_2, \bar{c}_3\}$ represents the cluster centres.

Algorithm 3: K-means clustering. Note: $d(x_i, x_j)$ is the distance between two elements.

Input : $X = \{x_1, x_2, \ldots, x_N\}$, number of centres (p), maximum number of iterations $(MaxIter)$, tolerance value (Tol)
Output: cluster centres $(\bar{C})$

Initialize the cluster centres $\bar{C} = \{\bar{c}_1, \bar{c}_2, \ldots, \bar{c}_p\}$;
for *iter from* 1 *to* $MaxIter$ **do**
    Assign each data to the nearest cluster: ;
    for i *from* 1 *to* p **do**
        $c_i = \{x_j : d(x_j, \bar{c}_i) \leq d(x_j, \bar{c}_l), j = 1, \ldots, N \text{ and } i \neq l\}$;
    Update the cluster centres ;
    for i *from* 1 *to* p **do**
        $\bar{c}_l = \frac{1}{|c_i|} \sum_{j=1}^{|c_i|} x_j, x_j \in c_i$
    if $(fval < Tol)$ **return**;
return;

Self-organization maps

Self-Organization Maps (SOM) is an unsupervised technique introduced by Kohonen in 1980s [100]. A SOM describes a mapping from a higher dimensional input space to a lower dimensional map space.

A SOM is an arrangement of p nodes or neurons into a regular spacing hexagonal/rectangular grid. Each node has associated a weight vector - same dimension as the input data vectors - and a position in the map space: $W = \{w_1, w_2, \ldots, w_p\}$, $w_i \in \Re^z$, z is the dimensionality of the input data. Each input data is fully connected to all nodes in the map, Fig. A.2.

At each step in the iterative process, the cluster unit with weights that best match the input pattern is selected as the winner and the winning unit and its neighbourhood are consequently updated. The iterative process in the Algorithm 4 describes the full process. In Chapter 3, Table 3.9 third column, can be seen an agricultural image partitioned with SOM algorithm.

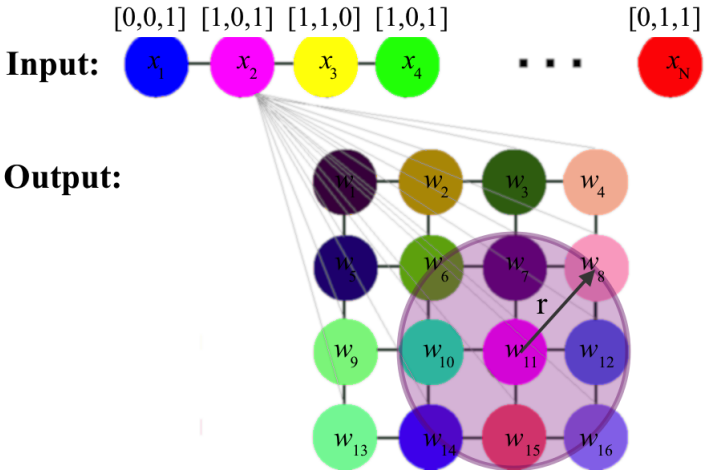


Figure A.2: Self-Organization Map.

Algorithm 4: Self-Organization Map.

Input : X, number of neurons (p), maximum number of iterations ($MaxEpoch$), radio size around the winner neuron (r), learning rate ($0 < \zeta < 1$)
Output: W

Initialize the p nodes in the map with random colour values $W = \{w_1, w_2, \ldots, w_p\}$;
for i *from* 1 *to* $MaxEpoch$ **do**
    Random input select: $x_i = X$;
    Find winner neuron: $w_l = \min_{j=1:p} d(x_i, w_j)$;
    Update weights in the range of the winner neuron;
    for j *from* 1 *to* $|\tilde{W}|$ **do**
        Smoothing kernel: $G_j = \zeta e^{-d(w_l, w_j/(2*\lambda))}$;
        Update weight: $w_j = w_l + G_j * (x_i - w_l)$;
return;

In our case, the input dataset consists of a set of image pixels X, where each element x_i represents spectral intensity of the pixel according to the tri-dimensional colour space, i.e. $x_i \in \Re^3$. Once SOM converges, the original image is mapped from a high colour space to a smaller colour space. The number of colours in this space is equal to the number of neurons in the SOM map. The final weight vectors form the new sample space. This set is used for clustering and allows determining a set of cluster centres.

The parameters involved in the Algorithm 4 are listed below:

- $d(x_i,x_j)$ is the Euclidean distance between two elements. If there is more than one with the same distance, then the winning weight is randomly chosen among the weights with the shortest distance.
- $\tilde{W}$ is the winner neuron $\tilde{W} \in W$.
- λ is the time constant.

- ζ is the learning rate factor which decreases monotonically with each iteration.
- r is the neighbourhood radio around $\tilde{W}$

Fuzzy C-means

Dunn initially introduced this method in 1973 [101]. It is a clustering method where an input data can belong to multiple clusters with different degrees of membership: $C = \{c_1, \ldots, c_p\}$, $c_i \in \Re^z$. The objective function to be minimized is given in A.2:

$$f_{cm} = \sum_{i=1}^{N} \sum_{j=1}^{p} \mu_{ij}^{m} (\parallel x_i - \bar{c}_j \parallel)^2 \tag{A.2}$$

where the exponent $m > 1$ controls the degree of fuzzy overlapping, c_j is the centre of the $j-$th cluster, and μ_{ij} is the membership degree of x_i in the $j-$th cluster. For each x_i, the sum of its membership values for all clusters is one. Algorithm 5 involves an iterative process for minimizing the above function. In Chapter 3, Table 3.9 fourth column, an agricultural illustrative image partitioned with the FCM algorithm can be seen.

Algorithm 5: Fuzzy-C means clustering.

Input : $X = \{x_1, x_2, \ldots, x_N\}$, number of centres ($p$), maximum number of iterations ($MaxIter$), minimum threshold value (Tol)
Output: cluster centres (C) and a the partition matrix (μ)

1 Randomly initialize the $p-$cluster membership values μ_{ij} ;
2 **for** j *from* 1 *to* $MaxIter$ **do**
3 Cluster centre: $c_j = \frac{\sum_{i=1}^{N} \mu_{ij}^{m} x - i}{\sum_{i=1}^{N} \mu_{ij}^{m}}$;
4 Fuzzy membership: $\mu_{ij} = \frac{1}{\sum_{k=1}^{p} \left(\frac{\| x_i - c_j \|}{\| x_i - c_k \|} \right)^{2/(m-1)}}$;
5 Objective function: $f_{val} = \sum_{i=1}^{N} \sum_{j=1}^{p} \mu_{ij}^{m} (\parallel x_i - \bar{c}_j \parallel)^2$
6 **if** $(fval < Tol)$ **return**;
7 **return**;

Graph-Based segmentation

This is a segmentation method proposed by Felzenzwalb and Huttenlocher [102]. Image segmentation is represented in terms of an undirected graph $G = (V, E)$ with vertices $v_i \in V$, the set of elements to be segmented, and edges $(v_i, v_j) \in E$ corresponding to pairs of neighbouring vertices. Each edge $(v_i, v_j) \in E$ has associated a corresponding weight

$w(v_i, v_j)$, which is a non-negative measure of the dissimilarity between neighbouring elements v_i and v_j. The elements in V are pixels and the weight of an edge is some measure of dissimilarity between the two connected pixels (e.g., the difference in intensity, colour, motion, or some other local attribute). In such work each pixel in the image is mapped to the feature space (x, y, r, g, b), where (x, y) is the location of the pixel in the image and (r, g, b) is the spectral colour value of the pixel. The Euclidean distance L_2 between points as the edge weights. The process is summarized in Algorithm 6.

Algorithm 6: Graph-Based segmentation algorithm. Note: The size of a component after a merge is the sum of the sizes of the two components.

Input : $G = (V, E)$, with n vertices and m edges.
Output: Segmentation of V into components $S = (C_1, \ldots, C_r)$

Sort E into $\Pi = (e_1, \ldots, e_m)$ by non-decreasing edge weight, e_q connects v_i and v_j ;
Set up S^0, each vertex v_i is its own component ;
for q *from* 1 *to* m **do**
    $C_i^{(q-1)}$ is the component of $S^{(q-1)}$ containing v_i ;
    $C_j^{(q-1)}$ is the component of $S^{(q-1)}$ containing v_j ;
    if $C_i^{q-1} \neq C_j^{q-1}$ *and* $w(e_q) \leq MInt(C_i^{q-1}, C_j^{q-1})$ **then**
        $S^q =$ merging C_i^{q-1}, C_j^{q-1} ;
    else
        $S^q = S^{q-1}$;
return $S = S^m$;

Authors provide a C++ implementation of the image segmentation algorithm. The program takes a colour image (ppm format) with the parameter values $(\hat{k}, \sigma', min)$ and produces the segmentation with a random colour assigned to each region. The resulting segmented image is in ppm format. The parameters are listed below:

a) Scale of observation $\hat{k}$: large $\hat{k}$ causes large components. For large images a suitable value is $\hat{k} = 300$.

b) Smoothing parameter σ': a Gaussian filter is used to smooth the image slightly before computing the edge weights, in order to compensate for digitization artifacts, default value $\sigma' = 0.8$.

c) Minimum size allowed of the components min.

Chapter 3, Table 3.9 fifth column, can be seen an image with over-segmentation applying the Algorithm A4.

Appendix B

Texture features

Tables B.1, B.2 and B.3 contain the mathematical expressions to compute different metrics used in Chapter 3 for texture descriptions. Table B.1 contains the first order statistics [127], Table B.2 metrics derived from the run length matrix [10], and finally Table B.3 summarizes the metrics derived from a co-occurrence matrix [103].

Feature	Expression	Feature	Expression
Mean		Entropy	
$\omega_\mu \stackrel{\text{def}}{=} \sum_{i=1}^{M} ih(i)$		$\omega_g \stackrel{\text{def}}{=} -\sum_{i=1}^{M} h(i)\log_2[h(i)]$	
Standard deviation		Skewness	
$\omega_\sigma \stackrel{\text{def}}{=} \sqrt{\sum_{i=1}^{M}(i-\omega_\mu)^2 h(i)}$		$\omega_s \stackrel{\text{def}}{=} \frac{1}{(\omega_\sigma)^3}\sum_{i=1}^{M}(i-\omega_\mu)^3 h(i)$	
Energy		Kurtosis	
$\omega_e \stackrel{\text{def}}{=} \sum_{i=1}^{M}[h(i)]^2$		$\omega_k \stackrel{\text{def}}{=} \frac{1}{(\omega_\sigma)^4}\sum_{i=1}^{M}(i-\omega_\mu)^4 h(i) - 3$	
Descriptor: $\rho_{fs} = \{\omega_\mu, \omega_\sigma, \omega_e, \omega_g, \omega_s, \omega_k\}$, dimension $z = 6$.			

Table B.1: First-order statistical properties for a normalized histogram (h) computed from the intensity values of an image or an interest region into the image. M is the number of grey level intensities, from 0 to 255.

Feature	Expression	Feature	Expression
Short run emphasis	$p_{sre} \stackrel{\text{def}}{=} \frac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}\frac{p(i,j)}{j^2}$	High grey-level run emphasis	$p_{hgre} \stackrel{\text{def}}{=} \frac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}p(i,j)i^2$
Long run emphasis	$p_{lre} \stackrel{\text{def}}{=} \frac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}p(i,j)j^2$	Short run low grey-level	$p_{srlge} \stackrel{\text{def}}{=} \frac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}\frac{p(i,j)}{i^2j^2}$
Grey-level non-uniformity	$p_{gln} \stackrel{\text{def}}{=} \frac{1}{n_r}\sum_{i=1}^{M}\left(\sum_{j=1}^{N}p(i,j)\right)^2$	Short run high grey-level	$p_{srhge} \stackrel{\text{def}}{=} \frac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}\frac{p(i,j)i^2}{j^2}$
Run-length non-uniformity	$p_{rln} \stackrel{\text{def}}{=} \frac{1}{n_r}\sum_{i=1}^{N}\left(\sum_{j=1}^{M}p(i,j)\right)^2$	Long run low grey-level emphasis	$p_{lrlge} \stackrel{\text{def}}{=} \frac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}\frac{p(i,j)j^2}{i^2}$
Run percentage	$p_{rpc} \stackrel{\text{def}}{=} n_r/n_p$	Long run grey-level emphasis	$p_{lrlge} \stackrel{\text{def}}{=} \frac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}p(i,j)i^2j^2$
Low grey-level run emphasis	$p_{lgre} \stackrel{\text{def}}{=} \frac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}\frac{p(i,j)}{i^2}$		

Descriptor: $\rho_{rlcm} = \{p_{sre}, p_{lre}, p_{gln}, p_{rln}, p_{rpc}, p_{lgre}, p_{hgre}, p_{srlge}, p_{srhge}, p_{lrlge}, p_{lrlge}\}$, dimension $z = 11$.

Table B.2: Run length matrix p: n_r total number of runs. n_p is the number of pixels on the window/image/region. For a run-length matrix, M is the number of grey level intensities, from 0 to 255, and N is the maximum run length, four directions were considered here [10].

,

Feature	Expression	Feature	Expression
Contrast		Entropy	
$g_c \stackrel{\text{def}}{=} \sum_{i=1}^{N_g-1} n^2 \left\{ \sum_{\lvert i-j\rvert=n} \frac{GCM(i,j)}{R} \right\}$		$g_e \stackrel{\text{def}}{=} \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} -\ln(GCM(i,j)) \times GCM(i,j)$	
Dissimilarity		Correlation	
$g_d \stackrel{\text{def}}{=} \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} GCM(i,j)\lvert i-j\rvert$		$g_{cc} \stackrel{\text{def}}{=} \frac{\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} ijGCM(i,j) - \mu_x\mu_y}{\sigma_x\sigma y}$	
Homogeneity		Inverse difference moment	
$g_h \stackrel{\text{def}}{=} \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \left(\frac{GCM(i,j)}{R}\right)^2$		$g_{idm} \stackrel{\text{def}}{=} \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \frac{GCM(i,j)}{1+(i-j)^2}$	
Angular second moment			
$g_{asm} \stackrel{\text{def}}{=} \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} GCM(i,j)^2$			
Descriptor: $\rho_{glcm} = \{g_c, g_d, g_h, g_{asm}, g_e, g_{cc}, g_{idm}\}$, dimension $z = 7$.			

Table B.3: Grey level Co-occurrence Matrix (GCM). Here μ_x, μ_y, σ_x and σ_y are the means and standard deviations of the marginal distributions associated with $GCM(i,j)/R$, R is a normalizing constant and $N_g = 256$ is the number of grey level intensities.

Vignetting correction

Process to minimize the vignetting effect in a colour image I of size $H \times W$. Chapter 2, Section 2.2.1.

Algorithm 7: Vignetting correction algorithm.

Input : I image of size $H \times W$ and the correction values (v_r, v_g, v_b)
Output: Corrected image

1 Image pattern centre: $(c_x, c_y) = (W/2, H/2)$
2 Pattern image: $P(x,y) = \sqrt{(x-c_x)^2 + (y-c_y)^2}$
3 Red channel correction: $R(x,y) = R(x,y) + v_r P(x,y) R(x,y)$
4 Green channel correction: $G(x,y) = G(x,y) + v_g P(x,y) G(x,y)$
5 Blue channel correction: $B(x,y) = B(x,y) + v_b P(x,y) B(x,y)$

Bibliography

[1] RHEA. Robot fleets for highly effective agriculture and forestry management. http://www.rhea-project.eu/, 2016.

[2] Andreas Koschan and Mongi Abidi. *Digital color image processing*. John Wiley & Sons, 2008.

[3] Patrick Fleischmann and Karsten Berns. A stereo vision based obstacle detection system for agricultural applications. In *Field and Service Robotics*, pages 217–231. Springer, 2016.

[4] Vincent Labatut and Hocine Cherifi. Accuracy measures for the comparison of classifiers. *CoRR*, abs/1207.3790, 2012. URL http://arxiv.org/abs/1207.3790.

[5] Yerania Campos, Humberto Sossa, and Gonzalo Pajares. Comparative analysis of texture descriptors in maize fields with plants, soil and object discrimination. *Precision Agriculture*, pages 1–19, 2016. ISSN 1573-1618. doi: 10.1007/s11119-016-9483-4. URL http://dx.doi.org/10.1007/s11119-016-9483-4.

[6] Wajahat Kazmi, Francisco Garcia, Jon Nielsen, Jesper Rasmussen, and Hans Andersen. Exploiting affine invariant regions and leaf edge shapes for weed detection. *Computers and Electronics in Agriculture*, 118:290 – 299, 2015. ISSN 0168-1699. doi: http://dx.doi.org/10.1016/j.compag.2015.08.023. URL http://www.sciencedirect.com/science/article/pii/S0168169915002495.

[7] Yerania Campos, Erik Rodner, Joachim Denzler, Humberto Sossa, and Gonzalo Pajares. *Vegetation Segmentation in Cornfield Images Using Bag of Words*, pages 193–204. Springer International Publishing, Cham, 2016. ISBN 978-3-319-48680-2. doi: 10.1007/978-3-319-48680-2_18. URL http://dx.doi.org/10.1007/978-3-319-48680-2_18.

[8] José Bengochea, Jesus Conesa, Dionisio Andújar, and Angela Ribeiro. Merge fuzzy visual servoing and gps-based planning to obtain a proper navigation behavior for a small crop-inspection robot. *Sensors*, 16(3):276, 2016.

[9] Arno Ruckelshausen. Bonirob, 2013. URL https://www.youtube.com/watch?v=x8Aq6qVQOkw.

[10] Xiaoou Tang. Texture information in run-length matrices. *IEEE transactions on image processing*, 7(11):1602–1609, 1998.

[11] Jan Behmann, Anne-Katrin Mahlein, Till Rumpf, Christoph Römer, and Lutz Plümer. A review of advanced machine learning methods for the detection of biotic stress in precision crop protection. *Precision Agriculture*, 16(3):239–260, 2014.

[12] Alex McBratney, Brett Whelan, Tihomir Ancev, and Johan Bouma. Future directions of precision agriculture. *Precision Agriculture*, 6(1):7–23, 2005. ISSN 1573-1618. doi: 10.1007/s11119-005-0681-8. URL http://dx.doi.org/10.1007/s11119-005-0681-8.

[13] Naiqian Zhang, Maohua Wang, and Ning Wang. Precision agriculture—a worldwide overview. *Computers and Electronics in Agriculture*, 36(2–3):113 – 132, 2002. ISSN 0168-1699. doi: http://dx.doi.org/10.1016/S0168-1699(02)00096-0. URL http://www.sciencedirect.com/science/article/pii/S0168169902000960.

[14] Emanuele Pierpaoli, Giacomo Carli, Erika Pignatti, and Maurizio Canavari. Drivers of precision agriculture technologies adoption: A literature review. *Procedia Technology*, 8:61 – 69, 2013. ISSN 2212-0173. doi: http://dx.doi.org/10.1016/j.protcy.2013.11.010. URL http://www.sciencedirect.com/science/article/pii/S2212017313000728.

[15] Ming Li, Kenji Imou, Katsuhiro Wakabayashi, and Shinya Yokoyama. Review of research on agricultural vehicle autonomous guidance. *International Journal of Agricultural and Biological Engineering*, 2(3):1–16, 2009.

[16] Shalal Nagham, Low Tobias, McCarthy Cheryl, and Hancock Nigel. A review of autonomous navigation systems in agricultural environments. *Innovative Agricultural Technologies for a Sustainable Futu*, pages 22–25, 2013.

[17] Hossein Mousazadeh. A technical review on navigation systems of agricultural autonomous off-road vehicles. *Journal of Terramechanics*, 50(3):211–232, 2013. ISSN 0022-4898. doi: http://dx.doi.org/10.1016/j.jterra.2013.03.004. URL http://www.sciencedirect.com/science/article/pii/S0022489813000220.

[18] Dionysis Bochtis, Claus Sørensen, and Patrizia Busato. Advances in agricultural machinery management: A review. *Biosystems engineering*, 126:69–81, 2014.

[19] Gonzalo Pajares, Iván García, Yerania Campos, Martín Montalvo, José Miguel Guerrero, Luis Emmi, Juan Romeo, María Guijarro, and Pablo Gonzalez-de Santos. Machine-vision systems selection for agricultural vehicles: A guide. *Journal of Imaging*, 2(4):34, 2016.

[20] VK Tewari, Ashok Kumar, Brajesh Nare, Satya Prakash, and Ankur Tyagi. Microcontroller based roller contact type herbicide applicator for weed control under row crops. *Computers and Electronics in Agriculture*, 104:40 – 45, 2014. ISSN 0168-1699. doi: http://dx.doi.org/10.1016/j.compag.2014.03.005. URL http://www.sciencedirect.com/science/article/pii/S0168169914000659.

[21] Su Hnin Hlaing and Aung Soe Khaing. Weed and crop segmentation and classification using area thresholding. *IJRET*, 3:375–382, 2014.

[22] Guoquan Jiang, Zhiheng Wang, and Hongmin Liu. Automatic detection of crop rows based on multi-rois. *Expert System Application*, 42(5):2429–2441, April 2015. ISSN 0957-4174. doi: 10.1016/j.eswa.2014.10.033. URL http://dx.doi.org/10.1016/j.eswa.2014.10.033.

[23] Wajahat Kazmi, Francisco Garcia, Jon Nielsen, Jesper Rasmussen, and Hans Andersen. Detecting creeping thistle in sugar beet fields using vegetation indices. *Computers and Electronics in Agriculture*, 112:10 – 19, 2015. ISSN 0168-1699. doi: http://dx.doi.org/10.1016/j.compag.2015.01.008. URL http://www.sciencedirect.com/science/article/pii/S0168169915000101. Precision Agriculture.

[24] Wenzhu Yang, Xiaolan Zhao, Sile Wang, Liping Chen, Xiangyang Chen, and Sukui Lu. *A New Approach for Greenness Identification from Maize Images*, chapter Intelligent Computing Theories and Methodologies: 11th International Conference, Fuzhou, China, August 2015, Proceedings, pages 339–347. Springer International Publishing, Cham, 2015. ISBN 978-3-319-22180-9. doi: 10.1007/978-3-319-22180-9_33. URL http://dx.doi.org/10.1007/978-3-319-22180-9_33.

[25] María Guijarro, Isabel Riomoros, Gonzalo Pajares, and Paula Zitinski. Discrete wavelets transform for improving greenness image segmentation in agricultural images. *Computers and Electronics in Agriculture*, 118:396–407, 2015.

[26] Moorthy Krishna, Moorthy Sruthi, Bernard Boigelot, and Benoî Mercatoris. *Effective segmentation of green vegetation for resource-constrained real-time*

applications, pages 257–266. Precision agriculture, 2015. doi: 10.3920/978-90-8686-814-8_31. URL http://www.wageningenacademic.com/doi/abs/10.3920/978-90-8686-814-8_31.

[27] Martín Montalvo, María Guijarro, José Miguel Guerrero, and Ángela Ribeiro. Identification of plant textures in agricultural images by principal component analysis. In *International Conference on Hybrid Artificial Intelligence Systems*, pages 391–401. Springer, 2016.

[28] Sajjad Yaghoubi, Negar Akbarzadeh, Shadi Bazargani, Sama Bazargani, Marjan Bamizan, and Maryam Asl. Autonomous robots for agricultural tasks and farm assignment and future trends in agro robots. *International Journal of Mechanical and Mechatronics Engineering IJMME-IJENS*, 13(03):1–6, 2013.

[29] Francisco Rovira, Ishani Chatterjee, and Verónica Sáiz. The role of gnss in the navigation strategies of cost-effective agricultural robots. *Computers and electronics in Agriculture*, 112:172–183, 2015.

[30] Anup Vibhute and SK Bodhe. Applications of image processing in agriculture: a survey. *International Journal of Computer Applications*, 52(2), 2012.

[31] Lalit Saxena and Leisa Armstrong. A survey of image processing techniques for agriculture. *Proceedings of Asian Federation for Information Technology in Agriculture*, 2014.

[32] Weis Martin and Roland Gerhards. Feature extraction for the identification of weed species in digital images for the purpose of site-specific weed control. *Precision agriculture*, 7:537–545, 2007.

[33] Youchun Ding, Du Chen, and Shumao Wang. The mature wheat cut and uncut edge detection method based on wavelet image rotation and projection. *African Journal of Agricultural Research*, 6(11):2609–2616, 2011.

[34] Kadim Nabeel and Salih Majjed. Environmental mobile robot based on artificial intelligence and visual perception for weed elimination. *Incas Bulletin*, 1(4): 11–25, 2012.

[35] José Miguel Guerrero, María Guijarro, Martin Montalvo, Juan Romeo, Luis Emmi, Angela Ribeiro, and Gonzalo Pajares. Automatic expert system based on images for accuracy crop row detection in maize fields. *Expert Systems with Applications*, 40(2):656–664, 2013.

[36] Ganesh Bhadane, Sapana Sharma, and Vijay Nerkar. Early pest identification in agricultural crops using image processing techniques. *International Journal of Electrical, Electronics and Computer Engineer-ing*, 2013.

[37] VK Tewari, Ashok Kumar Arudra, Satya Prakash Kumar, Vishal Pandey, and Narendra Singh Chandel. Estimation of plant nitrogen content using digital image processing. *Agricultural Engineering International: CIGR Journal*, 15(2): 78–86, 2013.

[38] McCarthy Cheryl, Hancock Nigel, and Raine Steven. Applied machine vision of plants: a review with implications for field deployment in automated farming operations. *Intelligent Service Robotics*, 3(4):209–217, 2010. ISSN 1861-2784. doi: 10.1007/s11370-010-0075-2. URL http://dx.doi.org/10.1007/s11370-010-0075-2.

[39] Chunhua Zhang and John Kovacs. The application of small unmanned aerial systems for precision agriculture: a review. *Precision Agriculture*, 13(6):693–712, 2012. ISSN 1573-1618. doi: 10.1007/s11119-012-9274-5. URL http://dx.doi.org/10.1007/s11119-012-9274-5.

[40] Vadivambal Rajagopal and Jayas Digvir. Applications of thermal imaging in agriculture and food industry: a review. *Food and Bioprocess Technology*, 4(2): 186–199, 2011. ISSN 1935-5149. doi: 10.1007/s11947-010-0333-5. URL http://dx.doi.org/10.1007/s11947-010-0333-5.

[41] Reyer Zwiggelaar, Christine Bull, and Michael Mooney. X-ray simulations for imaging applications in the agricultural and food industries. *Journal of Agricultural Engineering Research*, 63(2):161 – 170, 1996. ISSN 0021-8634. doi: http://dx.doi.org/10.1006/jaer.1996.0018. URL http://www.sciencedirect.com/science/article/pii/S0021863496900189.

[42] Sunil Mathanker, Weckler Paul, and Bowser Timothy. X-ray applications in food and agriculture: a review. *Transactions of the ASABE*, 56(3):1227–1239, 2013.

[43] Luiz Pires, Osny Bacchi, and Klaus Reichardt. Gamma ray computed tomography to evaluate wetting/drying soil structure changes. *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, 229(3–4):443 – 456, 2005. ISSN 0168-583X. doi: http://dx.doi.org/10.1016/j.nimb.2004.12.118. URL http://www.sciencedirect.com/science/article/pii/S0168583X04014132.

[44] Sebastian Haug, Michaels Andreas, Peter Biber, and Jorn Ostermann. Plant classification system for crop /weed discrimination without segmentation. In *IEEE Winter Conference on Applications of Computer Vision*, pages 1142–1149, March 2014. doi: 10.1109/WACV.2014.6835733.

[45] María Guijarro, Gonzalo Pajares, Isabel Riomoros, Javier Herrera, Burgos Artizzu, and Angela Ribeiro. Automatic segmentation of relevant textures in agricultural images. *Computers and Electronics in Agriculture*, 75(1):75 – 83, 2011. ISSN 0168-1699. doi: http://dx.doi.org/10.1016/j.compag.2010.09.013. URL http://www.sciencedirect.com/science/article/pii/S0168169910001924.

[46] John Rouse. Monitoring the vernal advancement and retrogradation (green wave effect) of natural vegetation. *Technical Report*, 1974.

[47] Wei Xiangqin, Jia Kun, Lan Jinhui, Li Yuwei, Zeng Yiliang, and Wang Chunmei. Automatic method of fruit object extraction under complex agricultural background for vision system of fruit picking robot. *Optik - International Journal for Light and Electron Optics*, 125(19):5684 – 5689, 2014. ISSN 0030-4026. doi: http://dx.doi.org/10.1016/j.ijleo.2014.07.001. URL http://www.sciencedirect.com/science/article/pii/S0030402614007426.

[48] Ohta Yu-Ichi, Kanade Takeo, and Sakai Toshiyuki. Color information for region segmentation. *Computer Graphics and Image Processing*, 13(3):222 – 241, 1980. ISSN 0146-664X. doi: http://dx.doi.org/10.1016/0146-664X(80)90047-7. URL http://www.sciencedirect.com/science/article/pii/0146664X80900477.

[49] Choi Keun, Han Sang, Han Sang, Park Kwang, Kim Kyung, and Kim Soohyun. Morphology-based guidance line extraction for an autonomous weeding robot in paddy fields. *Computers and Electronics in Agriculture*, 113:266 – 274, 2015. ISSN 0168-1699. doi: http://dx.doi.org/10.1016/j.compag.2015.02.014. URL http://www.sciencedirect.com/science/article/pii/S0168169915000563.

[50] Jorge Torres, Francisca López, and José Manuel Peña. An automatic object-based method for optimal thresholding in uav images: Application for vegetation detection in herbaceous crops. *Computers and Electronics in Agriculture*, 114:43 – 52, 2015. ISSN 0168-1699. doi: http://dx.doi.org/10.1016/j.compag.2015.03.019. URL http://www.sciencedirect.com/science/article/pii/S0168169915001052.

[51] Michael Schwarz, William Cowan, and John Beatty. An experimental comparison of rgb, yiq, lab, hsv, and opponent color models. *ACM Transactions on Graphics*, 6(2):123–158, April 1987. ISSN 0730-0301. doi: 10.1145/31336.31338. URL http://doi.acm.org/10.1145/31336.31338.

[52] David Woebbecke, Von Bargen, George Meyer, and David Mortensen. Color indices for weed identification under various soil, residue, and lighting conditions. *Transactions of the ASAE*, 38(1):259–269, 1995.

[53] George Meyer, T. Mehta, Michael Kocher, and Samal A. Textural imaging and discriminant analysis for distinguishing weeds for spot spraying. *Transactions of the ASAE*, 41(4):1189, 1998.

[54] Takashi Kataoka, Toshihiro Kaneko, and Hiroshi Okamoto. Crop growth estimation system using machine vision. In *Advanced Intelligent Mechatronics. Proceedings. IEEE/ASME International Conference on*, volume 2, pages b1079–b1083. IEEE, 2003.

[55] George Meyer and João Neto. Verification of color vegetation indices for automated crop imaging applications. *Computers and Electronics in Agriculture*, 63(2), 2008.

[56] David Woebbecke, George Meyer, Von Bargen, and David Mortensen. Plant species identification, size, and enumeration using machine vision techniques on near-binary images. In *Applications in Optical Science and Engineering*, pages 208–219. International Society for Optics and Photonics, 1993.

[57] Mahmood Golzarian and Ross Frick. Classification of images of wheat, ryegrass and brome grass species at early growth stages using principal component analysis. *Plant Methods*, 7(1):1–11, 2011. ISSN 1746-4811. doi: 10.1186/1746-4811-7-28. URL http://dx.doi.org/10.1186/1746-4811-7-28.

[58] Angela Ribeiro, Cesar Fernandez, Judit Barroso, and García Maria. Development of an image analysis system for estimation of weed. In *Proceedings of the 5th European Conf. On Precision Agriculture*, volume 5, pages 169–174, 2005.

[59] Meng Qingkuan, Qiu Ruicheng, He Jie, Zhang Man, Ma Xiaodan, and Liu Gang. Development of agricultural implement system based on machine vision and fuzzy control. *Computers and Electronics in Agriculture*, 112:128 – 138, 2015. ISSN 0168-1699. doi: http://dx.doi.org/10.1016/j.compag.2014.11.006. URL http://www.sciencedirect.com/science/article/pii/S0168169914002877. Precision Agriculture.

[60] Rafael Gonzalez and Richard Woods. *Digital Image Processing*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 3 edition, 2006. ISBN 013168728X.

[61] P. Balasubramaniam and V. Ananthi. Segmentation of nutrient deficiency in incomplete crop images using intuitionistic fuzzy c-means clustering algorithm. *Nonlinear Dynamics*, 83(1):849—866, 2015. ISSN 1573-269X.

[62] Mengni Ye, Zhiguo Cao, Zhenghong Yu, and Xiaodong Bai. Crop feature extraction from images with probabilistic superpixel markov random field.

Computers and Electronics in Agriculture, 114:247 – 260, 2015. ISSN 0168-1699. doi: http://dx.doi.org/10.1016/j.compag.2015.04.010. URL http://www.sciencedirect.com/science/article/pii/S0168169915001179.

[63] Beibei Cheng and Eric Matson. *Artificial Intelligence and Soft Computing: 14th International Conference, ICAISC, Zakopane, Poland, June 14-18, Proceedings*, chapter A Feature-Based Machine Learning Agent for Automatic Rice and Weed Discrimination, pages 517–527. Springer International Publishing, Cham, 2015. ISBN 978-3-319-19324-3. doi: 10.1007/978-3-319-19324-3_46. URL http://dx.doi.org/10.1007/978-3-319-19324-3_46.

[64] Santos Thiago, Koenigkan Luciano, Barbedo Jayme, and Rodrigues Gustavo. *Computer Vision ECCV 2014 Workshops: Zurich, Switzerland, Proceedings*, chapter 3D Plant Modeling: Localization, Mapping and Segmentation for Plant Phenotyping Using a Single Hand-held Camera, pages 247–263. Springer International Publishing, 2015. ISBN 978-3-319-16220-1. doi: 10.1007/978-3-319-16220-1_18. URL http://dx.doi.org/10.1007/978-3-319-16220-1_18.

[65] Radu Ionescu, Andreea Popescu, Marius Popescu, and Dan Popescu. Biomassid: A biomass type identification system for mobile devices. *Computers and Electronics in Agriculture*, 113:244 – 253, 2015. ISSN 0168-1699. doi: http://dx.doi.org/10.1016/j.compag.2015.03.002. URL http://www.sciencedirect.com/science/article/pii/S0168169915000599.

[66] Parekh Himani, Thakore Darshak, and Jaliya Udesang. A survey on object detection, classification and tracking methods. In *International Journal of Engineering Research and Technology. 11*, volume 3. ESRSA Publications, 2014.

[67] Joannes Oroko and GN Nyakoe. Obstacle avoidance and path planning schemes for autonomous navigation of a mobile robot: a review. In *Proceedings of Sustainable Research and Innovation Conference*, pages 314–318, 2014.

[68] Denis Fortun, Patrick Bouthemy, and Charles Kervrann. Optical flow modeling and computation: a survey. *Computer Vision and Image Understanding*, 134: 1–21, 2015.

[69] Haiyang Chao, Yu Gu, and Marcello Napolitano. A survey of optical flow techniques for uav navigation applications. In *Unmanned Aircraft Systems, ICUAS, International Conference on*, pages 710–716. IEEE, 2013.

[70] Changhong Chen, Jimin Liang, Heng Zhao, Haihong Hu, and Jie Tian. Frame difference energy image for gait recognition with incomplete silhouettes. *Pattern Recognition Letters*, 30(11):977–984, 2009.

[71] Chris Stauffer and Eric Grimson. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition. IEEE Computer Society Conference on.*, volume 2. IEEE, 1999.

[72] Ajay Mittal, Abdelaziz Bensrhair, and Edwin Hancock. Obstacle detection by means of stereo feature matching. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 1618–1622. IEEE, 2014.

[73] Payal Panchal, Gaurav Prajapati, Savan Patel, Hinal Shah, and Jitendra Nasriwala. A review on object detection and tracking methods. *International Journal for Research in Emerging Science and Technology*, 2:7–12, 2015.

[74] Hooshang Mazinan. A powerful approach to real-time mobile objects tracking in crowded environments utilizing 3d video analysis of fixed camera, particle filter and neural network. *Optik-International Journal for Light and Electron Optics*, 125(21):6357–6365, 2014.

[75] Bastian Leibe, Konrad Schindler, Nico Cornelis, and Luc Van Gool. Coupled object detection and tracking from static cameras and moving vehicles. *IEEE transactions on pattern analysis and machine intelligence*, 30(10):1683–1698, 2008.

[76] Kinan Panchsheria and Mehul Parikh. A comprehensive survey on moving object segmentation methods. *International Journal For Technological Research In Engineering*, 2:2347–4718, 2015.

[77] Giulio Reina and Annalisa Milella. Towards autonomous agriculture: Automatic ground detection using trinocular stereovision. *Sensors*, 12(9):12405–12423, 2012.

[78] Peter Biber, Ulrich Weiss, Michael Dorna, and Amos Albert. Navigation system of the autonomous agricultural robot bonirob. In *Workshop on Agricultural Robotics: Enabling Safe, Efficient, and Affordable Robots for Food Production (Collocated with IROS 2012), Vilamoura, Portugal*, 2012.

[79] Miguel Garrido, Manuel Perez, Constantino Valero, Chris Gliever, Bradley Hanson, and David Slaughter. Active optical sensors for tree stem detection and classification in nurseries. *Sensors*, 14(6):10783–10803, 2014.

[80] Gonçalo Monteiro, Cristiano Premebida, Paulo Peixoto, and Urbano Nunes. Tracking and classification of dynamic obstacles using laser range finder and

vision. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–7, 2006.

[81] Bertrand Douillard, Dieter Fox, and Fabio Ramos. A spatio-temporal probabilistic model for multi-sensor object recognition. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2402–2408. IEEE, 2007.

[82] Cristiano Premebida, Oswaldo Ludwig, and Urbano Nunes. Lidar and vision-based pedestrian detection system. *Journal of Field Robotics*, 26(9): 696–711, 2009.

[83] Patrick Ross, Andrew English, David Ball, Ben Upcroft, Gordon Wyeth, and Peter Corke. Novelty-based visual obstacle detection in agriculture. In *IEEE International Conference on Robotics and Automation, ICRA*, pages 1699–1705. IEEE, 2014.

[84] Sharon Nissimov, Jacob Goldberger, and Victor Alchanatis. Obstacle detection in a greenhouse environment using the kinect sensor. *Computers and Electronics in Agriculture*, 113:104–115, 2015.

[85] A Talukder, R Manduchi, A Rankin, and L Matthies. Fast and reliable obstacle detection and segmentation for cross-country navigation. In *Intelligent Vehicle Symposium*, volume 2, pages 610–618. IEEE, 2002.

[86] Caio César Mendes, Fernando Osório, and Denis Wolf. An efficient obstacle detection approach for organized point clouds. In *Intelligent Vehicles Symposium, IV*, pages 1203–1208. IEEE, 2013.

[87] Jeng Wei, Francisco Rovira, John Reid, and Shufeng Han. Obstacle detection using stereo vision to enhance safety of autonomous machines. *Transactions of ASAE*, 48(6):2389–2397, 2005.

[88] Fernando Cheein, G Steiner, Paina Perez, and Ricardo Carelli. Optimized eif-slam algorithm for precision agriculture mapping based on stems detection. *Computers and electronics in agriculture*, 78(2):195–207, 2011.

[89] Gokhan Bayar, Marcel Bergerman, and Ahmet Koku. Improving the trajectory tracking performance of autonomous orchard vehicles using wheel slip compensation. *Biosystems Engineering*, 2016.

[90] Jose Gázquez, Nuria Castellano, and Francisco Manzano. Intelligent low cost telecontrol system for agricultural vehicles in harmful environments. *Journal of Cleaner Production*, 113:204–215, 2016.

[91] Kim Steen, Peter Christiansen, Henrik Karstoft, and Rasmus Jørgensen. Using deep learning to challenge safety standard for highly autonomous machines in agriculture. *Journal of Imaging*, 2(1):6, 2016.

[92] Yuanjie Zheng, Stephen Lin, Chandra Kambhamettu, Jingyi Yu, and Sing Kang. Single-image vignetting correction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12):2243–2256, Dec 2009. ISSN 0162-8828. doi: 10.1109/TPAMI.2008.263.

[93] Juan Romeo, José Miguel Guerrero, Martín Montalvo, Luis Emmi, María Guijarro, Pablo Gonzalez-de Santos, and Gonzalo Pajares. Camera sensor arrangement for crop/weed detection accuracy in agronomic images. *Sensors*, 13 (4):4348–4366, 2013.

[94] José Miguel Guerrero. *Vision system for precision agriculture: real-time crop rows and weeds detection in maize fields*. PhD thesis, University Complutense of Madrid, 2015.

[95] Luis Emmi, Mariano Gonzalez, Gonzalo Pajares, and Pablo Gonzalez. Integrating sensory/actuation systems in agricultural vehicles. *Sensors*, 14(3): 4014–4049, 2014.

[96] Yerania Campos, Humberto Sossa, and Gonzalo Pajares. Maize image dataset, 2016. URL https://www.fdi.ucm.es/profesor/pajares/ACIVS/.

[97] Christopher Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. ISBN 0387310738.

[98] Nikhil Pal and Sankar Pal. A review on image segmentation techniques. *Pattern recognition*, 26(9):1277–1294, 1993.

[99] James MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297, Berkeley, California, 1967. University of California Press. URL http://projecteuclid.org/euclid.bsmsp/1200512992.

[100] Teuvo Kohonen, editor. *Self-organizing Maps*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1997. ISBN 3-540-62017-6.

[101] Joseph Dunn. A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. *Journal of Cybernetics*, 3(3):32–57, 1973. doi: 10.1080/01969727308546046.

[102] Pedro Felzenszwalb and Daniel Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.

[103] Robert Haralick and Karthikeyan Shanmugam. Textural features for image classification. *Transactions on systems, man, and cybernetics*, 3(6):610–621, 1973.

[104] David Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[105] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features. *Computer vision and image understanding*, 110(3):346–359, 2008.

[106] Gerard Salton and Michael McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA, 1986.

[107] Anna Bosch, Xavier Muñoz, and Robert Martí. Which is the best way to organize/classify images by content. *Image and Vision Computing*, 25(6):778 – 791, 2007. ISSN 0262-8856. doi: http://dx.doi.org/10.1016/j.imavis.2006.07.015. URL http://www.sciencedirect.com/science/article/pii/S0262885606002253.

[108] Li Fei-Fei and Pietro Perona. A bayesian hierarchical model for learning natural scene categories. In *Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 524–531. IEEE, 2005.

[109] Chih-Wei Hsu and Chih-Jen Lin. A comparison of methods for multiclass support vector machines. *IEEE transactions on Neural Networks*, 13(2):415–425, 2002.

[110] Bernhard Schölkopf and Alexander Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2002.

[111] Jerome Friedman. Another approach to polychotomous classification. Technical report, Department of Statistics, Stanford University, 1996.

[112] Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2 (3):27, 2011. URL http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[113] Andrea Vedaldi and Brian Fulkerson. Vlfeat open source library, 2008. URL http://www.vlfeat.org/.

[114] Clemens Brust, Sven Sickert, Marcel Simon, Erik Rodner, and Joachim Denzler. Convolutional patch networks with spatial prior for road detection and urban

scene understanding. *CoRR*, abs/1502.06344, 2015. URL http://arxiv.org/abs/1502.06344.

[115] Clemens Brust, Sven Sickert, Marcel Simon, Erik Rodner, and Joachim Denzler. Cn24 code, 2015. URL https://github.com/cvjena/cn24.

[116] Björn Fröhlich, Erik Rodner, and Joachim Denzler. *Computer Vision - ACCV 2012: 11th Asian Conference on Computer Vision, Daejeon, Korea, November 5-9*, chapter Semantic Segmentation with Millions of Features: Integrating Multiple Cues in a Combined Random Forest Approach, pages 218–231. Springer Berlin Heidelberg, 2013. ISBN 978-3-642-37331-2. doi: 10.1007/978-3-642-37331-2_17. URL http://dx.doi.org/10.1007/978-3-642-37331-2_17.

[117] Yerania Campos, Humberto Sossa, and Gonzalo Pajares. Spatio-temporal analysis for obstacle detection in agricultural videos. *Application Soft Computing*, 45(C):86–97, August 2016. ISSN 1568-4946. doi: 10.1016/j.asoc.2016.03.016. URL http://dx.doi.org/10.1016/j.asoc.2016.03.016.

[118] Gonzalo Pajares. A hopfield neural network for image change detection. *Transactions on Neural Networks*, 17(5):1250–1264, September 2006. ISSN 1045-9227. doi: 10.1109/TNN.2006.875978. URL http://dx.doi.org/10.1109/TNN.2006.875978.

[119] Javier Arroyo, María Guijarro, and Gonzalo Pajares. An instance-based learning approach for thresholding in crop images under different outdoor conditions. *Computers and Electronics in Agriculture*, 127:669 – 679, 2016. ISSN 0168-1699. doi: http://dx.doi.org/10.1016/j.compag.2016.07.018. URL http://www.sciencedirect.com/science/article/pii/S0168169916305130.

[120] Stig Larsson and Vidar Thomé. *Partial differential equations with numerical methods*, volume 45. Springer Science and Business Media, 2008.

[121] Edward Dougherty, Roberto Lotufo, and The International Society for Optical Engineering. *Hands-on morphological image processing*, volume 71. SPIE Optical Engineering Press Washington, 2003.

[122] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *Automatica*, 11(285-296):23–27, 1975.

[123] Yong Yang and Shuying Huang. Image segmentation by fuzzy c-means clustering algorithm with a novel penalty term. *Computing and Informatics*, 26(1):17–31, 2012.

[124] John Barron, David Fleet, and Steven Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43—77, 1994. ISSN 1573-1405.

[125] Diane Larlus, Jakob Verbeek, and Frédéric Jurie. Category level object segmentation by combining bag-of-words models with dirichlet processes and random fields. *International Journal of Computer Vision*, 88(2):238–253, 2010. ISSN 1573-1405. doi: 10.1007/s11263-009-0245-x. URL http://dx.doi.org/10.1007/s11263-009-0245-x.

[126] Jean-François Lalonde, Alexei Efros, and Srinivasa Narasimhan. Detecting ground shadows in outdoor consumer photographs. In *Proceedings of the 11th European Conference on Computer Vision: Part II*, ECCV'10, pages 322–335, Berlin, Heidelberg, 2010. Springer-Verlag. ISBN 3-642-15551-0, 978-3-642-15551-2. URL http://dl.acm.org/citation.cfm?id=1888028.1888053.

[127] Andrzej Materka and Michal Strzelecki. Texture analysis methods–a review. *Technical university of lodz, institute of electronics, COST B11 report, Brussels*, pages 9–11, 1998.